Adaptive Multiscale Approaches to Regression and Trend Segmentation

Hyeyoung Maeng

Department of Statistics

London School of Economics and Political Sciences

This dissertation is submitted for the degree of

Doctor of Philosophy

December 2019

To Jusin

Declaration

I certify that the thesis I have presented for examination for the MPhil/PhD degree of the London School of Economics and Political Science is solely my own work other than where I have clearly indicated that it is the work of others (in which case the extent of any work carried out jointly by me and any other person is clearly identified in it).

 I declare that my thesis consists of 81109 words.

I confirm that Chapters 3 and 4 were jointly co-authored with Professor Piotr Fryzlewicz and I contributed 80% of these works.

Chapter 3 has been published as Maeng, H. and Fryzlewicz, P. (2019). Regularised forecasting via smooth-rough partitioning of the regression coefficients. *Electronic Journal of Statistics*, 13, 2093-2120.

Chapter 4 has been submitted to a peer-reviewed statistical journal and we plan to submit Chapter 5 for publication soon.

Hyeyoung Maeng

December 2019

Acknowledgements

First I would like to thank my supervisor, Professor Piotr Fryzlewicz, for his heartfelt support, unbounded patience and extremely invaluable guidance over the past few years. It has been a great joy and privilege to work with such a fine researcher who has endless ideas, immense knowledge and enthusiasm. I am deeply grateful for his time and effort spent on my work.

I am also very grateful for the financial support from the London School of Economics Statistics PhD Scholarship. Thanks also to all the staff and students in the Department of Statistics at the London School of Economics for providing such a great working environment.

I would like to thank my parents and sister for their continual encouragement and unconditional love throughout my life. I am also deeply indebted to my husband Jusin for his consistent support, love and humour, there are no words to say how grateful I am.

Finally, to my loving God, who provided me with the opportunity to continue my research and whose love and compassion throughout this process enabled me to get through it smoothly.

Abstract

Data-adaptive modelling has enjoyed increasing popularity across a wide range of statistical problems. This thesis studies three adaptive multiscale approaches, one in regression and two in trend segmentation.

We first introduce a way of modelling temporal dependence in random functions, assuming that those random curves are discretised on an equispaced grid. Considering a common dependence structure across the discretised curves, we predict the most recent point from the past observations in the framework of linear regression. Our model partitions the regression parameters into a smooth and a rough regime where rough regression parameters are used for observations located close to the response variable while the set of regression coefficients for the predictors positioned far from the response variable are assumed to be sampled from a smooth function. The smoothness change-point and the regression parameters are jointly estimated, and the asymptotic behaviour of the estimated change-point is presented. The performance of our new model is illustrated through simulations and four real data examples including country fertility data, pollution data, stock volatility series and sunspot number data.

Secondly, we study the detection of multiple change-points corresponding to linear trend changes or point anomalies in one-dimensional data. We propose a data-adaptive multiscale decomposition of the data through an unbalanced wavelet transform, hoping that the sparse representation of the data is achieved through this decomposition. The entire procedure consists of four steps and we provide a precise recipe of each.

We show that the performance of our method is particularly remarkable in detecting point anomalies or frequent change-points with short segments. The consistency of the estimated number and locations of change-points is investigated, and the practicality of our approach is demonstrated through simulations and real data examples including Iceland temperature data and sea ice extent of the Arctic and the Antarctic.

Lastly, we introduce a new model for detecting trend changes in high-dimensional panel data which is an extension of the one-dimensional multiscale approach described above into the high-dimensional settings. We investigate two scenarios, change-points in piecewise-constant and piecewise-linear signals. The new approach performs well across a wide range of signals, not only when the changes occur in most of the data sequences but also when only a sparse subset of data sequences changes. The consistency of the estimated number and locations of change-points is shown under two scenarios considered. The usefulness of our approach is demonstrated through numerical studies and two real data examples, South Africa temperature data and sea ice extent of the Arctic and the Antarctic.

Table of contents

List of figures 11

List of tables 14

1 Introduction 16

2 Literature review 20

2.1 Regularisations on the functional linear regression coefficient 20

2.1.1 Functional linear regression 21

2.1.2 Finding null subregions via variable selection techniques 24

2.1.3 Selecting predictive design points 25

2.1.4 Change-point detection ideas 26

2.1.5 Partial functional linear regression 27

2.2 Change-point detection in one-dimensional data 28

2.2.1 Segmentation of piecewise-constant signal 29

2.2.2 Segmentation of piecewise-linear signal 38

2.3 Change-point detection in panel data 43

2.3.1 Early works for the case of single change-point 44

2.3.2 Multiple change-point detection in multivariate time series . . . 47

2.3.3 High-dimensional change-point problem 49

3 Smooth-Rough Partitioning of the regression coefficients 54

3.1 Introduction . . . 54

3.2 Model and its estimation . . . 60

3.2.1 Joint estimation procedure for parameters . . . 63

3.2.2 Selection of the tuning parameters . . . 67

3.3 Theoretical results . . . 68

3.4 Simulations . . . 72

3.4.1 Competing methods . . . 73

3.4.2 Simulation results . . . 75

3.5 Data applications . . . 80

3.5.1 Country fertility rate data . . . 80

3.5.2 Nitrogen oxides in Mexico City . . . 83

3.5.3 High frequency volatility series . . . 85

3.5.4 Monthly numbers of sunspots . . . 88

3.6 Proofs . . . 90

3.6.1 Proof of Theorem 3.1 . . . 91

4 Trend Segmentation in data sequences 96

4.1 Introduction . . . 96

4.2 Methodology . . . 101

4.2.1 Summary of TrendSegment . . . 101

4.2.2 TGUW transformation . . . 102

4.2.3 Thresholding . . . 110

4.2.4 Inverse TGUW transformation . . . 113

4.2.5 Post-processing for consistency of change-point detection . . . 113

4.2.6 Extra discussion of TGUW transformation . . . 116

4.3 Theoretical results . . . 123

4.4 Simulation study . . . 127
4.4.1 Parameter choice . . . 127
4.4.2 Simulation settings . . . 128
4.4.3 Competing methods and estimators . . . 128
4.4.4 Results . . . 130
4.5 Data applications . . . 138
4.5.1 Average January temperatures in Iceland . . . 138
4.5.2 Monthly average sea ice extent of Arctic and Antarctic . . . 141
4.6 Proofs . . . 143
4.6.1 Some useful lemmas . . . 143
4.6.2 Proof of Theorems 4.1 - 4.3 . . . 150

5 Trend Segmentation for high-dimensional panel data 154
5.1 Introduction . . . 154
5.2 Methodology . . . 158
5.2.1 Settings . . . 158
5.2.2 Structure of HiTS . . . 159
5.2.3 HiTGUW transformation . . . 160
5.2.4 Thresholding . . . 177
5.2.5 Inverse HiTGUW transformation . . . 178
5.2.6 Post-processing for consistent estimation . . . 179
5.3 Theoretical results . . . 182
5.3.1 Under temporal independence and general cross-sectional dependence . . . 182
5.3.2 Under a specific form of temporal dependence and general cross-sectional dependence . . . 186

5.3.3 Under temporal independence and a specific form of cross-sectional dependence . . . 187
5.4 Simulations . . . 188
5.4.1 Parameter choice . . . 188
5.4.2 Simulation setting . . . 190
5.4.3 Competing methods . . . 192
5.4.4 Simulation results . . . 192
5.5 Data applications . . . 209
5.5.1 Average January temperatures in South Africa . . . 209
5.5.2 Monthly average sea ice extent of Arctic and Antarctic . . . 212
5.6 Proofs . . . 215
5.6.1 Some useful lemmas . . . 215
5.6.2 Proof of Theorems 5.1 - 5.5 . . . 222
5.6.3 Proof of Corollaries 5.1 - 5.4 . . . 228

6 Conclusions 230

References 233

List of figures

2.1 Construction of tree for the example in Section 2.2.1. 36

3.1 The daily curves of hourly average nitrogen oxides in Mexico City in 2016. 57

3.2 Square-rooted monthly numbers of sunspots and its PACF and ACF. . 58

3.3 The example of the estimated regression coefficients of the daily curves of nitrogen oxides level. 61

3.4 Mean of $\{M(q)\}_{1\leq q\leq 50}$ (left) and $\{SIC(q)\}_{1\leq q\leq 50}$ over 100 simulation runs for Case 2. 66

3.5 True regression parameters of Cases 1-4. 73

3.6 Mean of $SIC(q)$ and barplots of the $\hat{q}$ over 100 simulation runs. 76

3.7 Barplots of the $\hat{q}$ estimated by minimising $SIC(q)$ in formula (3.7). . . 76

3.8 True and estimated regression parameters for Cases 1-4. 77

3.9 The fertility rates at age 20 from 1974 to 2009 for 31 countries. 81

3.10 Barplots of the $\hat{q}_1$ for MLR, $\hat{q}_2$ for $\text{SRP}_{\mathcal{L}}$ and $\hat{q}$ for $\text{SRP}_{\mathcal{C}}$ for the case study in Section 3.5.1. 82

3.11 A randomly selected estimated regression coefficients of the six parametric methods for predicting fertility rates at age 20. 82

3.12 Barplots of the $\hat{q}_1$ for MLR, $\hat{q}_2$ for $\text{SRP}_{\mathcal{L}}$ and $\hat{q}$ for $\text{SRP}_{\mathcal{C}}$ for the case study in Section 3.5.2. 84

3.13 A randomly selected estimated regression coefficients of the six parametric methods for predicting the average of nitrogen oxides level. 84
3.14 Barplots of the $\hat{q}_1$ for MLR, $\hat{q}_2$ for $\text{SRP}_{\mathcal{L}}$ and $\hat{q}$ for $\text{SRP}_{\mathcal{C}}$ for the case study in Section 3.5.3. 86
3.15 A randomly selected estimated regression coefficients of the six parametric methods for predicting closing volatility of the Disney stock. 87
3.16 Estimated regression coefficients of the six parametric methods for predicting the sunspot number. 89

4.1 January average temperature in Reykjavik recorded from 1763 to 2013. 97
4.2 Construction of tree for the example in Section 4.2.2. 104
4.3 Diagrams of the "connected" rule and the "two together" rule. 111
4.4 Examples of data with its underlying signal studied in Section 4.4. . . . 129
4.5 Change-point analysis for January average temperature in Reykjavik. . 139
4.6 Change-point analysis for January average temperature in Reykjavik with varying constants in the threshold of the ID procedure. 140
4.7 Change-point analysis for January average temperature in Reykjavik with varying constants in the threshold of the CPOP procedure. 140
4.8 Change-point analysis for January average temperature in Reykjavik with varying constants in the threshold of the TrendSegment procedure. 141
4.9 Change-point analysis for the monthly average sea ice extent of the Arctic in February. 144
4.10 Change-point analysis for the monthly average sea ice extent of the Arctic in September. 145
4.11 Change-point analysis for the monthly average sea ice extent of the Antarctic in February. 146

4.12 Change-point analysis for the monthly average sea ice extent of the Antarctic in September. 147

5.1 January average temperature curves of 50 cities in South Africa. 157

5.2 Construction of tree for the example of scenario (S1) in Section 5.2.3. . 162

5.3 Construction of tree for the example of scenario (S2) in Section 5.2.3. . 168

5.4 Examples of data with its underlying signal studied in scenario (S1). . . 195

5.5 Examples of data with its underlying signal studied in scenario (S2). . . 205

5.6 Change-point analysis for the data in Section 5.5.1. 210

5.7 The post-thresholded HiTS estimates for the temperature data in South Africa. 212

5.8 The geographical locations of 50 cities in South Africa classified into four categories by the post-thresholding of the HiTS algorithm. 213

5.9 The monthly sea ice extent in the Arctic and the HiTS estimate. 214

5.10 The post-thresholded HiTS estimates for the sea ice extent in the Arctic. 215

5.11 The monthly sea ice extent in the Antarctic and the HiTS estimate. . . 216

5.12 The post-thresholded HiTS estimates for the sea ice extent in the Antarctic. 217

List of tables

3.1 The mean(sd) of SSE over 100 simulation runs. 77

3.2 The mean(sd) of MSPE over 100 simulation runs. 78

3.3 The percentages indicating how many time-points are selected as the most-predictive design points by MPDP. 79

3.4 The summary of 100 MSPE's ($\times 10^6$) for the case study in Section 3.5.1. 81

3.5 The summary of 100 MSPE's ($\times 10^2$) for the case study in Section 3.5.2. 85

3.6 The summary of 100 MSPE's for the case study in Section 3.5.3. 87

3.7 MSPE ($\times 10^2$) for the case study from Section 3.5.4. 90

4.1 Notation. See Section 4.2.2 for formulae for the terms listed. 103

4.2 Simulation results for models (M1)-(M4) for the Gaussian error. 132

4.3 Simulation results for models (M5)-(M8) for the Gaussian error. 133

4.4 Simulation results for models (M1)-(M4) for $\epsilon_t \sim AR(1)$. 134

4.5 Simulation results for models (M5)-(M8) for $\epsilon_t \sim AR(1)$. 135

4.6 Simulation results for models (M1)-(M4) for $\epsilon_t \sim t_5$. 136

4.7 Simulation results for models (M5)-(M8) for $\epsilon_t \sim t_5$. 137

5.1 Notation. See Section 5.2.3 for formulae for the terms listed. 161

5.2 Simulation results for models (M1)-(M6) in the "complete-overlap" case with $n = 100$ in scenario (S1). 196

5.3 Simulation results for models (M1)-(M6) in the "complete-overlap" case with $n = 300$ in scenario (S1). 197
5.4 Simulation results for models (M1)-(M6) in the "complete-overlap" case with $n = 500$ in scenario (S1). 198
5.5 Simulation results for models (M1)-(M6) in the "half-overlap' case with $n = 100$ in scenario (S1). 199
5.6 Simulation results for models (M1)-(M6) in the "half-overlap' case with $n = 300$ in scenario (S1). 200
5.7 Simulation results for models (M1)-(M6) in the "half-overlap' case with $n = 500$ in scenario (S1). 201
5.8 Simulation results for models (M1)-(M6) in the "no-overlap" case with $n = 100$ in scenario (S1). 202
5.9 Simulation results for models (M1)-(M6) in the "no-overlap" case with $n = 300$ in scenario (S1). 203
5.10 Simulation results for models (M1)-(M6) in the "no-overlap" case with $n = 500$ in scenario (S1). 204
5.11 Simulation results for models (M1)-(M6) in the "complete-overlap" case in scenario (S2). 206
5.12 Simulation results for models (M1)-(M6) in the "half-overlap" case in scenario (S2). 207
5.13 Simulation results for models (M1)-(M6) in the "no-overlap" case in scenario (S2). 208
5.14 50 cities in South Africa classified into four categories by the post-thresholding of the HiTS algorithm. 211

Chapter 1

Introduction

In many applications, statistical models are often designed to capture some changes that the ingredient of a model undergoes. Changes arise in many contexts such as changes in distribution of a time series or jumps in a sequence of regression coefficients where jump is regarded as a type of change. When such change occurs at some points, detecting those change-points is not only an important task but also useful for a higher-level representation of the data that is taken as a follow-up analysis of the change-point detection. It is indeed a problem of significant interest in many application and recent examples include detecting price inflation (Groen et al., 2013), detection of DNA copy number variants (Olshen et al., 2004), detecting change-points in functional magnetic resonance imaging (fMRI) data (Cribben and Yu, 2017), climate change detection (Robbins et al., 2011) and detecting exoplanets from light curve data (Fisch et al., 2018).

The main body of this thesis deals with the problem of detecting a single or multiple change-points where the changes occur in a sequence of regression coefficients or in univariate or high-dimensional data sequences. The core methodologies introduced in Chapters 3-5 are all data-adaptive and view the change-point detection as a multiscale problem, where a methodology is referred to as data-adaptive if it can adjust the

parameters or the order of its optimisation process to the data at hand. In Chapter 2, we provide a literature review on the relevant fields including various regularisations imposed on the functional linear regression coefficient and change-point detection methodologies for univariate and high-dimensional data sequences. The remainder of this thesis is structured as follows.

Chapter 3. Smooth-Rough Partitioning of the regression coefficients

In this chapter, we propose the Smooth-Rough Partition (SRP) model, a new way of modelling temporal dependence in random functions. Assuming the curves are discretised on an equispaced grid, the most recent points are predicted from the past observations in the framework of linear regression. The proposed model reflects the 'decaying memory' structure of the time series by partitioning the regression parameters into a smooth and a rough regime. Specifically, unconstrained (rough) regression parameters are used for observations located close to the response variable, while the set of regression coefficients for the predictors positioned far from the response variable are assumed to be sampled from a smooth function. The regression parameters and the point at which the change in smoothness occurs are jointly estimated from the data, and the asymptotic behaviour of the estimated change-point is analysed. We illustrate its good performance through simulations. The usefulness of partitioning the effects into two scales is demonstrated through four real datasets, one of which shows that the SRP framework can also be a useful alternative to the AR modelling especially when the time series possesses long-term dependence. The SRP model is implemented in the R package `srp`, available from CRAN.

Chapter 4. Trend Segmentation in data sequences

In this chapter, we propose TrendSegment, a new methodology for detecting multiple change-points corresponding to linear trend changes or point anomalies in one-dimensional data. A core ingredient of TrendSegment is a new Tail-Greedy Unbalanced Wavelet (TGUW) transform: a conditionally orthonormal, bottom-up transformation of the data through an adaptively constructed unbalanced wavelet basis, which results in a sparse representation of the data. The bottom-up nature of this multiscale decomposition enables the detection of point anomalies and linear trend changes at once as the decomposition focuses on local features in its early stages and on global features next. The proposed method merges multiple regions in a single pass over the data which not only reduces the computational complexity but also guarantees the consistency of the estimated number and locations of change-points under the assumption of i.i.d. Gaussian noise. We demonstrate the practicality of our approach through simulations and two real data examples, involving Iceland temperature data and sea ice extent of the Arctic and the Antarctic. Our methodology is available from the R package `trendsegmentR`.

Chapter 5. Trend Segmentation for high-dimensional panel data

As an extension of TrendSegment introduced in Chapter 4 into high-dimensional settings, we propose a new methodology for detecting trend changes in high-dimensional panel data which is referred to as High-dimensional Trend Segmentation (HiTS). The key ingredient of the HiTS procedure is a high-dimensional version of the TGUW transform proposed in Chapter 4, that constructs an unbalanced wavelet basis (which is common to all univariate data sequences) in a data-adaptive way, by performing consecutive merges of neighbouring regions from bottom to top. We in-

vestigate HiTS in two scenarios, one of which is the case when the set of underlying signals are all piecewise-constant and the other case is for piecewise-linear signals. Our methodology is designed to be robust in estimating the number and locations of change-points not only when the changes are dense across the panel but also when the changes occur only in a sparse subset of the coordinates. We consider both independent and dependent noise settings and show the consistency of the estimated number and locations of change-points under two scenarios considered. The HiTS procedure is easy to implement and rapidly computed even in the case of a large number of coordinates. The usefulness of HiTS is demonstrated through extensive numerical studies and two real data examples including South Africa temperature data and sea ice extent of the Arctic and the Antarctic. The new methodology is implemented in our GitHub repository (Maeng, 2019c).

We note that each of the main chapters includes their own introduction section where more detailed motivations are given. Finally, Chapter 6 gives a brief summary of the contributions of this thesis and points a number of possible directions for future research.

Chapter 2

Literature review

In this chapter, we provide a literature review on the adaptive multiscale approaches studied in this thesis. This involves change-point detection in regression parameters and trend segmentation in low- and high-dimensional settings.

2.1 Regularisations on the functional linear regression coefficient

In this section, we introduce some existing approaches in the literature that are relevant to our proposal in Chapter 3. These mainly cover regularisations imposed on the functional linear regression coefficient by detecting a change-point or by finding informative regions of the regression parameter. We first briefly introduce the scalar-on-function regression in Section 2.1.1 then review the relevant methodologies in later sections. In Chapter 3, the important differences between those methodologies and our proposal will be highlighted and the performances are also compared and contrasted.

2.1.1 Functional linear regression

Over the last few decades, functional data analysis (FDA) has been growing in importance and enjoying increased attention where an extensive review can be found in Ramsay and Silverman (2005). Functional objects arise in many contexts and the applications in the literature include prediction of daily curves of particulate matter in the air (Aue et al., 2015), testing stationarity of intraday price curves of a financial asset (Horváth et al., 2014), modelling the dynamics of fertility rate (Chen et al., 2017), studying the effect of air pollution on the mortality rate across cities (Kong et al., 2016), prediction of the protein content of meat from spectral curves (Zhu et al., 2014), investigation of a bike sharing system by predicting bike pick-up counts (Han et al., 2018), choosing predictive days from daily egg-laying counts for fruit flies (Ji and Müller, 2017) and predicting sucrose content of orange juice from its near-infrared spectrum (Ferraty et al., 2010).

The main ingredients of functional data analysis are random functions $X_i \in L^2[0,1]$ where $i = 1, \ldots, n$ and $[0,1]$ is a compact subset of $\mathbb{R}$. If the random functions X_i are believed to possess temporal dependence and are analysed by separating the domain they live on into shorter units, we call such a data structure functional time series. Functional time series analysis has been an active field of research in recent years. The best-known model in this area is the first-order functional autoregressive model proposed by Bosq (2000). Other recent contributions include testing for stationarity (Horváth et al., 2014), testing for mean functions in a two-sample problem (Horváth et al., 2013), testing for error correlation (Gabrys et al., 2010) and prediction (Antoniadis et al., 2006; Aue et al., 2015).

On the other hand, if the functions are used as a predictor for explaining a scalar response variable Y, this simply describes the standard functional linear regression:

$$Y_i = \mu + \int_0^1 \beta(t)X_i(t)dt + \varepsilon_i, \quad i = 1, ..., n, \tag{2.1}$$

where $\beta(t) \in L^2([0,1])$ is a square integrable function, $X(t)$ is a functional covariate, μ is a scalar coefficient and ε is a random error with mean zero and finite variance. This model has been widely studied in the literature e.g. the reader can find a review of numerous approaches to scalar-on-function regression in Reiss et al. (2017).

Our interest is in the coefficient function $\beta(t)$ which shows the relationship between Y and $X(t)$ where the interpretation is fairly straightforward; the subintervals with greater $|\beta(t)|$ is where X is more influential to predict Y. Since $X(t)$ has infinite dimension, Y can be perfectly predicted unless any restriction is imposed on $\beta(t)$. The required regularisation on $\beta(t)$ is usually achieved by a basis expansion, which enables a finite number of basis functions to approximate the infinite-dimensional function. In general, the basis functions can be classified into two categories: 1) predetermined basis vectors such as the Fourier series, splines or wavelets and 2) data-driven basis vectors, mostly eigenfunctions obtained from the functional principal component analysis, where more details can be found in Ramsay and Silverman (2005). In what follows, we give a brief description of each case.

Fixed basis functions

When $X(t)$ is assumed to be fully observed and $\beta(t)$ is spanned by a number of basis functions as $\beta(t) \approx \sum_{l=1}^{L} b_l B_l(t)$, the integration term in (2.1) can be approximated as

$$\int_0^1 X_i(t)\beta(t)dt \approx \sum_{l=1}^{L} \left\{ \int_0^1 X_i(t)B_l(t)dt \right\} b_l. \tag{2.2}$$

The regularisation usually take one of two possible ways: 1) choosing an appropriate size of L to prevent both undersmoothing and oversmoothing of $\beta(t)$ and 2) adding roughness penalty which controls the smoothness of $\beta(t)$ under the fixed size of L that is large enough to avoid undersmoothing. The latter is often called 'penalised splines' due to the penalty term in its objective function and can be viewed as a generalised ridge regression. The detailed estimation procedure of the latter is presented in Section 3.2.1.

Data-driven basis functions

When the data-driven basis functions are used, it is common to assume that the unknown $\beta(t)$ belongs to the function space of $X(t)$. This enables us to expand both $\beta(t)$ and $X(t)$ with orthonormal eigenfunctions, $\psi_1, \psi_2, ...$, of the integral operator Γ with kernel k where the singular value decomposition of the covariance function is defined as

$$k(t,s) = \text{cov}(X(t), X(s)) = \sum_{j=1}^{\infty} v_j \psi_j(t) \psi_j(s),$$

and $(\Gamma\phi)(t) = \int k(t,s)\phi(t)ds$ with a square integrable function $\phi(t)$. The eigenfunctions are obtained from the functional principal component analysis and both $\beta(t)$ and $X(t)$ can be written as

$$\beta(t) = \sum_{k=1}^{\infty} b_k \psi_k(t), \quad X_i(t) = \mu_x(t) + \sum_{k=1}^{\infty} a_{ik} \psi_k(t), \tag{2.3}$$

where $\mu_x(t)$ is the mean curve of X. Thanks to the orthonormality of eigenfunctions, the integration term in (2.1) is now simplified as

$$\int_0^1 X_i(t)\beta(t)dt = \sum_{k=1}^{\infty} a_{ik} b_k, \tag{2.4}$$

and this allows us to write the model (2.1) as $Y_i = \mu + \sum_{k=1}^{\infty} a_{ik} b_k + \varepsilon_i$. In practice, we usually use the truncated version, $Y_i = \mu + \sum_{k=1}^{L} a_{ik} b_k + \varepsilon_i$, where a scree plot is often engaged for choosing an optimal L. When the functional principal component scores, $\{a_{ik}\}_{k=1}^{L}$, are predicted under a fixed L, the regression coefficients, $\{b_k\}_{k=1}^{L}$, can be simply estimated through the standard least-squares estimation.

2.1.2 Finding null subregions via variable selection techniques

As a way of regularising the standard scalar-on-function regression coefficient, we can consider finding subregions or points in the regression function over which the changes in the corresponding $X(t)$ have a greater effect on the response variable. From this point of view, some researchers have used ideas from variable selection to obtain $\beta(t) = 0$ for the non-informative subintervals and $\beta(t) \neq 0$ for the informative ones.

Functional linear regression that's interpretable

James et al. (2009) employ the LASSO (Tibshirani, 1996) and the Dantzig selector (Candes and Tao, 2007) with the aim of improving the interpretability of $\beta(t)$ in (2.1). They assume sparsity in the d^{th} derivative of $\beta(t)$, for example if the model has the sparsity conditions, $d = 0, 2$, then the estimator $\hat{\beta}(t)$ would be a mix of zero regions (returned by the condition $d = 0$, i.e. sparsity in the 0th derivative) and regions of linear trend (guaranteed by $d = 2$, i.e. the sparsity in the second derivative). Dividing the time period into a fine grid of points, they use the variable selection methods to determine whether each grid point of $\beta(t)$ has zero d^{th} derivative. In practice, they adopt two derivatives, $d = 0$ (as a default) and the other chosen from $d = 2, 3, 4$ by minimising cross validation (CV) error. As the smoothness of $\hat{\beta}(t)$ only depends on the non-zero d, this approach is not designed to reflect a varying smoothness behaviour in $\beta(t)$.

Other approaches

Similarly, Zhou et al. (2013) use the Dantzig selector and the SCAD approach (Fan and Li, 2001) and Lin et al. (2015) propose a functional version of SCAD by combining the SCAD method and smoothing splines to obtain a smooth and sparse estimator for the functional coefficient.

2.1.3 Selecting predictive design points

Another way of regularising the functional linear regression coefficient is finding a set of grid points on the given interval in which $X(t)$ has the greatest predictive impact on Y. We examine the relevant methodologies in what follows.

Point of impact

Kneip et al. (2016) introduce the following model under the name of *functional linear regression model with points of impact*:

$$Y_i = \sum_{j=1}^{q} \alpha_j X_i(t_j) + \int_0^1 \beta(t) X_i(t) dt + \varepsilon_i, \quad i = 1, \ldots, n, \tag{2.5}$$

where both functional and scalar parameters are explored in the framework of scalar-on-function regression. The model is proposed for some situations when only one or several points in $X(t)$ have a significant relevance on the scalar response variable Y. In estimating the locations of the influential points, they remove the observations adjoining the points of impact. After collecting the candidates of points of impact under a suitable cut-off parameter, the model parameters q, α and β in (2.5) are estimated by minimising the Schwarz's Information Criterion (SIC, Schwarz (1978)). Similar studies include McKeague and Sen (2010) who explore the selection of a single point of impact with the motivation from gene expression data.

Most-predictive design points

Ferraty et al. (2010) propose an explicit way of choosing a few influential points $(t_1, ..., t_r)$ in the following functional nonparametric model,

$$Y_i = m(X_i(t_1), ..., X_i(t_r)) + \varepsilon_i, \quad i = 1, ..., n, \tag{2.6}$$

where m is a smooth functional that would be estimated to capture a nonlinear relationship after choosing r predictive design points, $X_i(t_1), ..., X_i(t_r)$, from the functional predictor $X_i(t)$. Finding several predictive design points is connected with the idea of reducing the infinite dimension of the functional covariate X to a lower dimension. The approach is based on the discretised curves rather than a full function X and the given value of r is assumed to be significantly smaller than the number of discrete observations of X. The set of predictive points is selected through the stepwise algorithm.

2.1.4 Change-point detection ideas

In this section, we introduce two methodologies including the idea of detecting a single change-point in the functional linear regression coefficient.

Truncation in $\beta(t)$

Hall and Hooker (2016) find the truncation point θ under the following truncated functional linear model:

$$Y_i = \mu + \int_0^{\theta} \beta(t) X_i(t) dt + \varepsilon_i, \quad i = 1, ..., n. \tag{2.7}$$

Under the full functional framework, the truncation point θ defines the non-zero interval in $\beta(t)$. The optimal θ is estimated from the entire interval $[0, 1]$ by minimising the

penalised least-squares as follows:

$$\hat{\theta} = \arg\min_{\theta} \sum_{i=1}^{n} \left\{ Y_i - \breve{\mu} - \int_0^{\theta} \breve{\beta}(t) X_i(t) dt \right\}^2 + n\lambda\theta^2, \tag{2.8}$$

where $(\breve{\mu}, \breve{\beta}(t))$ are the pilot estimators obtained without a truncation constraint and λ is the tuning parameter adjusting the location of $\hat{\theta}$ closer to the lower endpoint of the interval. This approach is motivated by a real example modelling particulate matter emissions (PM) from diesel trucks.

Discontinuity in $\beta(t)$

Other works based on the partitioning idea include Goia and Vieu (2015). While Hall and Hooker (2016) engage one continuous function $\beta(t)$ for fitting both non-zero and zero regions, Goia and Vieu (2015) use two smooth functions, $\beta_1(t)$ and $\beta_2(t)$, by dividing the entire interval into two subintervals with one discontinuity point. They suggest the partitioned functional single index model as follows:

$$Y_i = \mu + g_1\left(\int_{[0,\lambda]} \beta_1(t) X_i(t) dt \right) + g_2\left(\int_{(\lambda,1]} \beta_2(t) X_i(t) dt \right) + \varepsilon_i, \quad i = 1, ..., n, \tag{2.9}$$

where g_1 and g_2 are smooth functions to be estimated and the breakpoint λ identifies a discontinuity in the functional regression coefficient.

2.1.5 Partial functional linear regression

The skeleton of our new model in Chapter 3 is similar to that of partial functional linear regression,

$$Y_i = \mu + \boldsymbol{\alpha}^{\mathrm{T}} \boldsymbol{Z}_i + \int_0^1 \beta(t) X_i(t) dt + \varepsilon_i, \quad i = 1, \ldots, n, \tag{2.10}$$

where Y is a scalar response variable, $\boldsymbol{Z}$ is a q-dimensional vector of scalar random variables and $X(t)$ is a functional random variable. This regression model was recently studied by Kong et al. (2016), Zhou et al. (2016), Zhou and Chen (2012), Shin and Lee (2012), Shin (2009), Aneiros-Pérez and Vieu (2008) and Goia (2012).

2.2 Change-point detection in one-dimensional data

We consider a univariate time series that is a collection of observations recorded in time order. Time series arise in many contexts, for example economics (stock price, unemployment rate, GDP, inflation, exchange rate), environment (pollution, temperature, precipitation, sea level, earthquake, wind speed, sea ice cover), medical sciences (DNA copy number, fMRI scans, brain activity records via EEG) and astronomy (counts of sunspots, light curves, satellite orbital cycles).

Changes in time series can be classified as distributional (e.g. mean or variance) change or trend (e.g. constant or linear or quadratic) change. Detecting the number and locations of distributional changes is important when the stationarity assumption is violated and the underlying process changes their distribution over time as it approximates the stationary time intervals by identifying their boundaries. On the other hand, detecting changes in trend can be useful for feature extraction or data mining as it reduces the dimension by dividing a time series into a number of pieces corresponding to features of interest. In both cases, segmenting time series is an important investigation in the initial stage of analysis as it can affect the analysis performed in later stages.

Especially, multiple change-point detection is a problem of importance in many applications; recent examples include automatic detection of change-points in cloud data to maintain the performance and availability of an app or a website (James et al., 2016), climate change detection in tropical cyclone records (Robbins et al., 2011), detecting

exoplanets from light curve data (Fisch et al., 2018), detecting changes in the DNA copy number (Bardwell and Fearnhead, 2017; Jeng et al., 2012; Olshen et al., 2004), estimation of stationary intervals in potentially cointegrated stock prices (Matteson et al., 2013), estimation of change-points in multi-subject fMRI data (Robinson et al., 2010) and detecting changes in vegetation trends (Jamali et al., 2015).

Change-point detection approaches have a form of either offline or online. Offline (posteriori) change-point detection algorithms identify the change-points in a retrospective view by investigating all the observed data points at once. By contrast, online detection algorithms do not operate with a fixed-length sequence; instead, the observations are received and monitored sequentially over time.

In the following sections, we focus mainly on a posteriori multiple change-point analysis for piecewise-constant and piecewise-linear signal models. We consider the change-point model

$$X_t = f_t + \varepsilon_t, \quad t = 1, \ldots, T, \tag{2.11}$$

where f_t is a deterministic and unknown piecewise-polynomial signal and ε_t's are random errors with zero mean and constant variance. The signal contains N unknown change-points, $\eta_1, \eta_2, \ldots, \eta_N$, at which the features of interest in f_t undergo changes.

2.2.1 Segmentation of piecewise-constant signal

A large body of trend segmentation deals with the case when f_t in (2.11) is a piecewise-constant signal and its change-points $\eta_1, \eta_2, \ldots, \eta_N$ are formulated as follows,

$$f_t = \theta_\ell \;\text{ for }\; t \in [\eta_{\ell-1} + 1, \eta_\ell],\; \ell = 1, \ldots, N + 1, \tag{2.12}$$

where N is either known or unknown, $\theta_1, \ldots, \theta_{N+1} \in \mathbb{R}$ and $\theta_\ell \neq \theta_{\ell+1}$ for $\ell = 1, \ldots, N$. We remark that detecting changes in mean can be categorised into either distributional or trend change.

Constrained optimisation approaches

One of the major classes of multiple change-point detection methodologies is based on the minimisation of criterion function as follows:

$$\underset{\eta_1,\ldots,\eta_N}{\arg\min} \Big\{ L(X_t, \eta_1, \ldots, \eta_N) + pen(N, \eta_1, \ldots, \eta_N) \Big\}, \tag{2.13}$$

where $L(\cdot)$ is called loss or cost function that has a form of likelihood (or least-squares) and measures the fit of estimated value to the data, while $pen(\cdot)$ is the penalty added to prevent overfitting. Under the assumption of Gaussian noise, Yao and Au (1989) consider the least-squares estimators of change-point locations when the number of change-points is fixed. Considering the number of change-points as the dimension of a model, Yao (1988) uses the Schwarz's Information Criterion (SIC, Schwarz (1978)), also known as the Bayesian information criterion (BIC), under the Gaussian assumption to estimate N which gives the following optimisation problem,

$$\underset{\eta_1,\ldots,\eta_N}{\arg\min} \left[T \log \left\{ \frac{1}{T} \sum_{\ell=1}^{N+1} \sum_{t=\eta_{\ell-1}+1}^{\eta_\ell} \Big(X_t - \bar{X}(\eta_{\ell-1}+1, \eta_\ell) \Big)^2 \right\} + 2N \log T \right], \tag{2.14}$$

where $\bar{X}(\eta_{\ell-1}+1, \eta_\ell)$ is the mean of $X_{\eta_{\ell-1}+1}, \ldots, X_{\eta_\ell}$. Examples of a penalty that is linear in the number of change-points can be found in Lee (1995), Lavielle and Moulines (2000) and Boysen et al. (2009). For a penalty depending on both the number and the locations of change-points, see Pan and Chen (2006) and Zhang and Siegmund (2007). Lee (1997) and Frick et al. (2014) relax the Gaussian assumption on ε_t to exponential families. In particular, Frick et al. (2014) is shown to control the family-wise error rate

while Li et al. (2016) suggest an approach based on the control of the false discovery rate.

Those penalty-based optimisations are often criticised for its computational speed of at least $O(T^2)$. To overcome this issue, Killick et al. (2012) introduce the Pruned Exact Linear Time (PELT) algorithm and Rigaill (2015) proposes the pruned Dynamic Programming Algorithm (pDPA), where both methods achieve a linear computational cost in best-case scenarios but the speeds remain quadratic in the worst cases. As an extension, Maidstone et al. (2017b) introduce two algorithms, one of which is the Functional Pruning Optimal Partitioning (FPOP) that uses functional pruning technique of Rigaill (2015) to solve the penalised minimisation problem and always prunes more than PELT. Other variants include the Generalized Functional Pruning Optimal Partitioning (GFPOP) proposed by Hocking et al. (2018) and the Generalized Pruned Dynamic Programming Algorithm (GPDPA) introduced by Hocking et al. (2017). Tickle et al. (2018) pursue the computational improvement of some of those dynamic programming approaches based on parallel computing.

Binary Segmentation

Binary Segmentation (Vostrikova, 1981) has been widely used in multiple change-point detection as it is conceptually simple and easy to implement. It has a top-down character in that it searches the entire dataset in the initial step, and if any change-point is detected then the same procedure is repeated for two subintervals split by the detected change-point. In detecting a change-point, a test statistic, $\mathcal{C}_{p,q,r}(X)$, defined for any $1 \leq p \leq q \leq r \leq T$ is used in that a change-point in $[p, r]$ is defined as

$$q^* = \underset{p \leq q \leq r}{\arg\max} \, |\mathcal{C}_{p,q,r}(X)|, \quad \text{if } \max_{p \leq q \leq r} |\mathcal{C}_{p,q,r}(X)| > \lambda, \tag{2.15}$$

where λ is a pre-specified threshold. For example, in the implementation of the Binary Segmentation, Vostrikova (1981) uses the Cumulative Sum (CUSUM) statistic:

$$\mathcal{C}_{p,q,r}(X) = \sqrt{\frac{r-q}{(r-p+1)(q-p+1)}} \sum_{t=p}^{q} X_t - \sqrt{\frac{q-p+1}{(r-p+1)(r-q)}} \sum_{t=q+1}^{r} X_t, \quad (2.16)$$

which can be constructed from the fact that under the assumption of Gaussian noise with a constant and known variance, maximising the size of CUSUM statistic in (2.16) is equivalent to finding the maximum likelihood estimator for the piecewise-constant signal with a single change-point. The Binary Segmentation procedure continues to search the shorter segments to the left and the right of the detected change-point as long as the maximum size of CUSUM statistic exceeds the threshold and stops searching if no more change-points are detected.

Although Binary Segmentation is one of the most popular approaches in multiple change-points detection, as it finds a single change-point at each segment (i.e. fitting a best piecewise-constant function with a single change-point in the least-squares sense), it may fail to perform adequately if $[p, r]$ in (2.16) contains more than one true change-point. There have been a number of works in the literature which attempt to remedy this issue but keep the idea of Binary Segmentation, for example Circular Binary Segmentation (Olshen et al., 2004; Venkatraman and Olshen, 2007), Wild Binary Segmentation (WBS, Fryzlewicz (2014)), Narrowest-Over-Threshold (NOT, Baranowski et al. (2019)) and Wild Binary Segmentation 2 (Fryzlewicz, 2018a). In detail, the WBS methodology adds the random characteristics which enhance the ability of CUSUM estimator in detecting multiple change-points. In the initial stage, it compares the CUSUM statistics, $\mathcal{C}_{p',q,r'}(X)$, obtained from the randomly selected segments $[p', r']$ rather than using a global CUSUM statistic, $\mathcal{C}_{1,q,T}(X)$, computed on the entire dataset $X_1, \ldots, X_T$. Then, the overall maximiser of the entire collection of CUSUM statistics is chosen as a change-point only when it exceeds a pre-specified

threshold, i.e.

$$q^* = \underset{p' \le q \le r'}{\arg\max} |\mathcal{C}_{p',q,r'}(X)|, \quad \text{if } \max_q |\mathcal{C}_{p',q,r'}(X)| > \lambda, \tag{2.17}$$

where λ is a pre-specified threshold. If the first change-point is declared, the WBS algorithm repeats the same procedure in the left and the right of the change-point as done in the Binary Segmentation. The WBS algorithm can in principle be extended to the detection of other types of change e.g. changes in the second order structure of a time series (Korkas and Fryzlewicz, 2017), however there is a possibility that the chosen interval contains two or more change-points, in which case the algorithm may fail other than piecewise-constant signal. To ensure that at most one change-point exists in the selected segment, Baranowski et al. (2019) propose a multiple change-point detection device termed Narrowest-Over-Threshold (NOT), which focuses on the narrowest segment among those whose contrast exceeds a pre-specified threshold, thus a change-point is chosen as,

$$q^* = \underset{p',\, \arg\max_{p' \le q \le r'} |\mathcal{C}_{p',q,r'}(X)|,\, r'}{\arg\min} \{|r' - p'| : \max_q |\mathcal{C}_{p',q,r'}(X)| > \lambda\}, \tag{2.18}$$

where λ is a pre-specified threshold. The NOT approach enhances the ability of CUSUM estimator by investigating short segments containing only one change-point with a high probability. Based on those test statistics introduced in Baranowski et al. (2019), Anastasiou and Fryzlewicz (2019) propose Isolate-Detect (ID) approach that continuously searches data segments for changes by expanding those segments from the leftmost and the rightmost of the entire interval, which can be seen as a modified sliding window algorithm. Other methods related to the Binary Segmentation include Fryzlewicz (2007), which uses the discrete unbalanced Haar wavelet transform

for nonparametric function estimation and shows that Binary Segmentation can be interpreted in terms of the unbalanced Haar wavelet.

Binary Segmentation has also been popularly used for one-dimensional data outside the piecewise-polynomial segmentation. Fryzlewicz and Subba Rao (2014) use the Binary Segmentation algorithm for detecting change-points in the structure of an auto-regressive conditional heteroscedastic model and Cho and Fryzlewicz (2012) consider the locally stationary wavelet (LSW) time series model and estimate change-points in the second-order structure. Other related methodologies for high-dimensional time series will be reviewed in Section 2.3.

Bottom-up structure

Bottom-up procedures have rarely been used in change-point detection. Matteson and James (2014) use an agglomerative algorithm for hierarchical clustering in the context of change-point analysis. Messer et al. (2014) propose a multiple filter algorithm which detects change-points by searching from the smallest to the largest window sizes of a time series. Fryzlewicz (2018b) introduces the Tail-Greedy Unbalanced Haar (TGUH) transform, a bottom-up and data-adaptive transformation of univariate sequences that performs multiple change-point detection in the piecewise-constant signal. In the initial stage of the TGUH transform, the raw data are considered smooth coefficients, i.e. $(s_{1,1}, s_{2,2}, \ldots, s_{T,T}) = (X_1, X_2, \ldots, X_T)$, and it recursively updates the sequence of smooth coefficients by merging the local segments, i.e. applying local conditionally orthonormal transformations. To decide which pair of neighbouring regions should be merged next, we compare the corresponding detail-type coefficients, where the detail coefficient for merging two neighbouring smooth coefficients $s_{p,q}$ and $s_{q+1,r}$ is defined as,

$$d_{p,q,r} = \sqrt{\frac{r-q}{r-p+1}} s_{p,q} - \sqrt{\frac{q-p+1}{r-p+1}} s_{q+1,r}, \tag{2.19}$$

where $s_{p,r} = (r-p+1)^{-1/2} \sum_{s=p}^{r} X_s$ is always achieved as the algorithm progresses. We can simply show that the formula of the detail coefficient in (2.19) is equal to that of CUSUM statistic in (2.16). As the magnitude of the detail coefficient implies the strength of the corresponding local constancy, we sort the sequence $|d_{p,q,r}|$ in non-decreasing order and give priority in merging to a pair of smooth coefficients corresponding to the smallest detail coefficient.

We now provide a simple example of the TGUH transformation where the accompanying illustration is in Figure 2.1. This example shows single merges at each pass through the data, although it can be generalised into multiple passes through the data which is referred to as "tail-greediness". We refer to j^{th} pass through the data as scale j. Assume that we have the initial input $\boldsymbol{s}^0 = (X_1, X_2, \ldots, X_5)$, so that the complete TGUH transform consists of 4 merges. We now show 4 example merges one by one.

Scale $j = 1$. From the initial input $\boldsymbol{s}^0 = (X_1, \ldots, X_5)$, we consider 4 pairs (X_1, X_2), (X_2, X_3), (X_3, X_4), (X_4, X_5) and compute the size of the detail for each pair, where the formula can be found in (2.19). Suppose that (X_2, X_3) gives the smallest size of detail, $|d_{2,2,3}|$, then merge (X_2, X_3) through the orthogonal transformation formulated as follows:

$$\begin{pmatrix} s_{p,r} \\ d_{p,q,r} \end{pmatrix} = \begin{pmatrix} -b_{p,q,r} & a_{p,q,r} \\ a_{p,q,r} & b_{p,q,r} \end{pmatrix} \begin{pmatrix} s_{p,q} \\ s_{q+1,r} \end{pmatrix}, \quad i = 1, \ldots, n. \tag{2.20}$$

where $a_{p,q,r} = \sqrt{\frac{r-q}{r-p+1}}$ and $b_{p,q,r} = -\sqrt{\frac{q-p+1}{r-p+1}}$. Then update the data sequence into $\boldsymbol{s} = (X_1, s_{2,3}, d_{2,2,3}, X_4, X_5)$.

Scale $j = 2$. The possible pairs for next merging are $(X_1, s_{2,3})$, $(s_{2,3}, X_4)$, (X_4, X_5). Assume that (X_4, X_5) gives the smallest size of detail coefficient $|d_{4,4,5}|$ among the three candidates, then we merge them through the or-

thogonal transformation formulated in (2.20) and now update the sequence into $\boldsymbol{s} = (X_1, s_{2,3}, d_{2,2,3}, s_{4,5}, d_{4,4,5})$.

Scale $j = 3$. We now compare two candidates for merging, $(X_1, s_{2,3})$, $(s_{2,3}, s_{4,5})$. Suppose that $(s_{2,3}, s_{4,5})$ has the smallest size of detail; we merge this pair and update the data sequence into $\boldsymbol{s} = (X_1, s_{2,5}, d_{2,2,3}, d_{2,3,5}, d_{4,4,5})$.

Scale $j = 4$. The only available pair is now $(X_1, s_{2,5})$, thus we merge this and update the data sequence into $\boldsymbol{s} = (s_{1,5}, d_{1,1,5}, d_{2,2,3}, d_{2,3,5}, d_{4,4,5})$. The transformation is completed with the updated data sequence which contains $T - 1 = 4$ detail and 1 smooth coefficients.

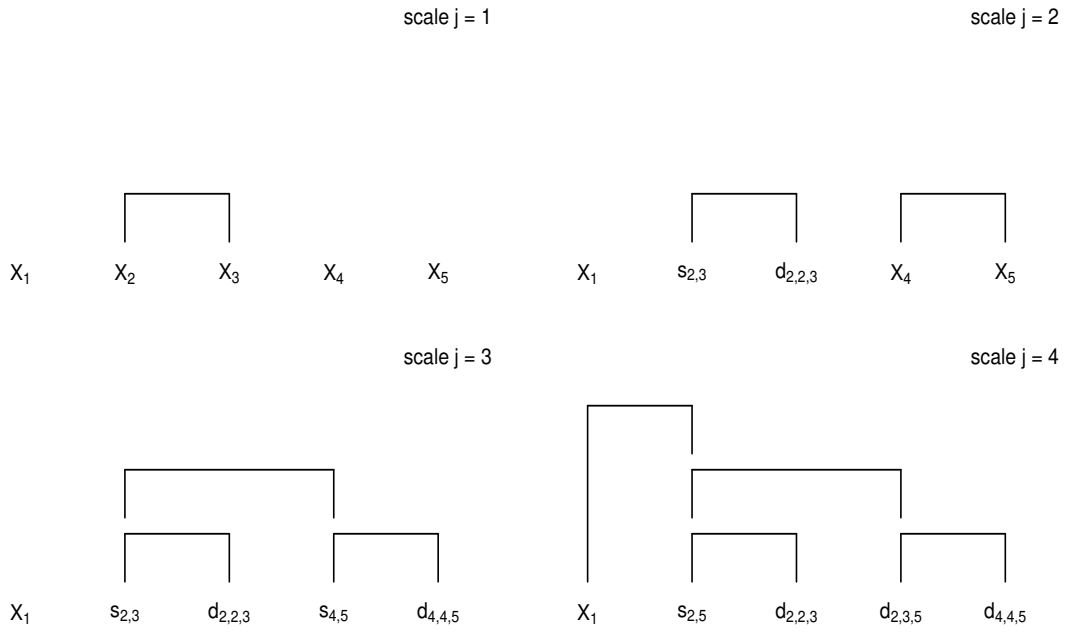


Fig. 2.1 Construction of tree for the example in Section 2.2.1; each diagram shows all merges performed up to the given scale.

One of the important properties of the TGUH transform is "tail-greediness" which reduces the computational complexity by allowing us to perform multiple merges over non-overlapping regions in a single pass over the data. It is called "tail-greedy" as

those details chosen in each pass correspond to the lower tail of their distribution. The resulting transformation of the data hopes to push the bulk of the variance of the input data vector in only a few detail-type coefficients arising at coarse levels, which enables the sparse representation of the data. In Section 4, we will introduce the Tail-Greedy Unbalanced Wavelet (TGUW) transform, that is an extension of the TGUH transform in Fryzlewicz (2018b) into the one for piecewise-linear signal settings.

Point anomalies

The detection of point anomalies has been widely studied in both time series and machine learning literature and an extensive review can be found in Chandola et al. (2009). There have been substantial discussions in statistical methodologies about how to achieve robustness to outliers, for example, in the framework of change-point analysis, Fearnhead and Rigaill (2019) propose to use a loss function that is less sensitive to the presence of outliers within a penalised optimisation framework. In contrast to the methodologies pursuing robustness to outliers, some methods are designed to detect anomalies. Fisch et al. (2018) propose an algorithm for detecting Collective And Point Anomalies (CAPA) with respect to mean and variance. The TrendSegment approach that will be introduced later in Chapter 4 also focuses on detecting point anomalies, but our framework is different from that of Fisch et al. (2018) in that we focus on linear trend changes and point anomalies in (2.11) with the underlying signal in (2.21), while their focus is not on trends but only on point and collective anomalies with respect to a constant baseline distribution.

Other approaches

There are a number of approaches which do not directly belong to any of the above categories, but we mention a few. Eichinger and Kirch (2018) study a moving sum

(MOSUM) procedure which detects a change-point by computing a statistic over sliding windows at a given bandwidth. In nonparametric settings, Harchaoui and Cappé (2007) and Harchaoui et al. (2009) propose kernel-based methods for multiple change-point detection and Haynes et al. (2017) investigate a nonparametric version of PELT method proposed by Killick et al. (2012). Lastly, although many Bayesian approaches are available in the literature, we only mention a selection. Early Bayesian methodologies for single change-point detection include Chernoff and Zacks (1964) and Broemeling (1972), and those for multiple change-points include Barry and Hartigan (1993), Inclan (1993) and Stephens (1994). More recent contributions include Bayesian inference for multiple change-point problems (Fearnhead, 2006; Fearnhead and Liu, 2007; Wilson et al., 2010) and detecting abnormal regions (Bardwell and Fearnhead, 2017).

2.2.2 Segmentation of piecewise-linear signal

Change-point detection in higher-order polynomial trends has recently attracted much attention in the literature and largely focuses on piecewise-linear segmentation. We consider the scenario in which the underlying signal f_t in (2.11) is formulated as follows,

$$f_t = \theta_{\ell,1} + \theta_{\ell,2}\, t, \quad \text{for} \quad t \in [\eta_{\ell-1}+1, \eta_\ell], \quad \ell = 1, \ldots, N+1, \tag{2.21}$$

where $\theta_{1,1}, \theta_{1,2}, \ldots, \theta_{N+1,1}, \theta_{N+1,2} \in \mathbb{R}$ and $f_{\eta_\ell} + \theta_{\ell,2} \neq f_{\eta_\ell+1}$ for $\ell = 1, \ldots, N$. This definition permits both continuous and discontinuous changes.

The change-point model in (2.11) with the piecewise-linear signal in (2.21) can be considered as a special case of segmented linear regression in which a sequence of T pairs of observations $(X_t, Y_t)_{t=1,\ldots,T}$ is segmented into a number of groups depending on the corresponding regression parameters and multiple regressions are performed on

those segments as follows:

$$Y_t = \theta_{\ell,1} + \theta_{\ell,2}\, X_t + \varepsilon_t, \quad \text{for} \quad t \in [\eta_{\ell-1}+1, \eta_\ell], \quad \ell = 1, \ldots, N+1, \tag{2.22}$$

where the change-point model in (2.11) with the underlying signal in (2.21) is obtained when $X_t = t$ in (2.22).

Single change-point

Early works on segmented linear regression in model (2.22) focus mainly on the single change-point case when $N = 1$ in (2.22). Quandt (1958) introduces a maximum likelihood method for detecting the unknown change-point and Smith and Cook (1980) propose a Bayesian analysis and use it for detecting the rejection time of transplanted kidneys. Worsley (1983) considers the piecewise multiple linear regression that is a generalised version of (2.22) with multiple regressors as follows:

$$Y_t = \theta_{\ell,1}\, X_{t,1} + \theta_{\ell,2}\, X_{t,2} + \cdots + \theta_{\ell,p}\, X_{t,p} + \varepsilon_t, \quad \text{for} \quad t \in [\eta_{\ell-1}+1, \eta_\ell], \quad \ell = 1, \ldots, N+1, \tag{2.23}$$

and studies the single change-point case when $N = 1$ in (2.23) by proposing a maximum likelihood method as an extension of Quandt (1958) designed for the model with one regressor.

Multiple change-points

Now we discuss relatively recent approaches proposed to deal with multiple change-points either in (2.22) or (2.23) under the penalised regression framework. Bai and Perron (1998) apply the least square principles to the model (2.23) and estimate the

locations of change-point by solving the following:

$$(\hat{\eta}_1, \ldots, \hat{\eta}_N) = \underset{\substack{\eta_1,\ldots,\eta_N \\ \eta_{\ell+1}-\eta_\ell \geq \beta}}{\arg\min} \left[\sum_{\ell=1}^{N+1} \sum_{t=\eta_{\ell-1}+1}^{\eta_\ell} \left\{ Y_t - (\hat{\theta}_{\ell,1}\, X_{t,1} + \hat{\theta}_{\ell,2}\, X_{t,2} + \cdots + \hat{\theta}_{\ell,p}\, X_{t,p}) \right\}^2 \right], \tag{2.24}$$

where N is assumed to be known, β is given and $(\hat{\theta}_{\ell,1}, \ldots, \hat{\theta}_{\ell,p})$ are the least-squares estimators for any fixed $(\eta_1, \ldots, \eta_N)$. As a practical solution of (2.24), Bai and Perron (2003) suggest an algorithm based on dynamic programming.

Under the continuity restriction at change-points, the model (2.22) can be reparameterised as

$$Y_t = \theta_{\eta_\ell} + \frac{\theta_{\eta_{\ell+1}} - \theta_{\eta_\ell}}{\eta_{\ell+1} - \eta_\ell}(t - \eta_\ell) + \varepsilon_t, \quad \text{for} \quad t \in [\eta_{\ell-1}+1, \eta_\ell], \quad \ell = 1, \ldots, N+1, \tag{2.25}$$

and under this framework, Maidstone et al. (2017a) consider l_0 penalty and propose a pruned dynamic programming algorithm to solve the minimisation problem,

$$\underset{\substack{N \\ \eta_1,\ldots,\eta_N \\ \theta_{\eta_1},\ldots,\theta_{\eta_{N+2}}}}{\arg\min} \left[\sum_{\ell=1}^{N+1} \left\{ \frac{1}{\sigma^2} \sum_{t=\eta_{\ell-1}+1}^{\eta_\ell} \left(Y_t - \theta_{\eta_\ell} - \frac{\theta_{\eta_{\ell+1}} - \theta_{\eta_\ell}}{\eta_{\ell+1} - \eta_\ell}(t - \eta_\ell) \right)^2 + \gamma(\eta_{\ell+1} - \eta_\ell) \right\} + \beta N \right], \tag{2.26}$$

where β is a positive penalty constant, $\gamma(\cdot)$ is a non-negative and non-decreasing function for penalising segment-length and σ^2 is assumed to be known.

Similarly, under the least-squares principle with the continuity constraint at change-points, Kim et al. (2009) and Tibshirani (2014) consider 'trend filtering' with the l_1 penalty, in which case the optimisation problem for the model (2.11) with the signal (2.21) is formulated as

$$\underset{f_t}{\arg\min} \left[(1/2) \sum_{t=1}^{T} (X_t - f_t)^2 + \lambda \sum_{t=2}^{T-1} |f_{t-1} - 2f_t + f_{t+1}| \right], \tag{2.27}$$

where λ is a positive tuning parameter. The trend filtering focuses on function estimation rather than change-point detection, however as studied in Lin et al. (2017), those two goals are connected to each other in the sense that a fast enough l_2 error rate of estimated function implies that each true change-point has an estimator nearby.

The Narrowest-Over-Threshold (NOT, Baranowski et al. (2019)) method and the Isolate-Detect (ID, Anastasiou and Fryzlewicz (2019)) approach introduced in Section 2.2.1 explicitly deal with higher-order polynomial trends e.g. piecewise-linear, by applying an appropriate test statistic that is derived from the generalised likelihood ratio under the assumption of Gaussian noise. When the underlying signal is a piecewise-constant function, the test statistic (often referred to as a contrast function) obtained from the generalised likelihood ratio is equivalent to the CUSUM statistic in (2.16), however in the case of higher-order polynomial, the contrast function has a more complicated form. Apart from the form of the contrast function, both NOT and ID algorithms operate in the same ways described in Section 2.2.1; NOT prefers the narrowest segment among those whose contrast exceeds a pre-specified threshold and continues to search in the Binary Segmentation framework and ID continuously searches expanding data segments for changes.

Piecewise-linear segmentation is an important investigation in the initial stage of analysis, thus often used in time series data mining. Keogh et al. (2004) mention that sliding windows, top-down and bottom-up approaches are three principal categories which most time series segmentation algorithms can be grouped into. Keogh et al. (2004) apply those three approaches to the detection of changes in linear trends in 10 different signals and discover that the performance of bottom-up methods is better than that of top-down methods and sliding windows, notably when the underlying signal has jumps, sharp cusps or large fluctuations. Their bottom-up algorithm merges adjacent segments of the data according to a criterion involving the minimum residual

sum of squares (RSS) from a linear fit, until the RSS falls under a certain threshold. However, the lack of precise recipes for the choice of this threshold parameter causes the performance of this method to be somewhat unstable, as we report in Section 4.4.

In nonparametric regression analysis, there have been some discussions about the estimation of jump regression curves. Jump regression can be related to change-point analysis in that both focus on curve segmentation, although the ultimate goals are different; the former is finding the discontinuous points in the regression curve and fitting an arbitrary continuous curve between any two consecutive jump points, the latter is estimating the number and locations of change-points in features of interest (e.g. changes in mean or slope). Early works in jump regression are established on the assumption that the number of jumps is known, for example kernel-type estimators of jump points (McDonald and Owen, 1986) and detecting jumps and sharp cusps by discrete wavelet transform (Wang, 1995). Under the assumption of the unknown number of jumps, Xia and Qiu (2015) propose a jump information criterion for optimising the number and sizes of jumps.

Finally, we discuss a few other approaches for multiple change-point detection. McZgee and Carleton (1970) suggest a hierarchical clustering-based approach, Kim et al. (2000) use several permutation tests with continuity constraint at change-points and Yu et al. (2007) propose a weighted least-squares approach for multiple change-point detection as an extension of Hudson (1966) that is designed for detecting a single change-point. Ertel and Fowlkes (1976) consider the piecewise multiple linear regression with multiple regressors and multiple change-points, which is the case when $N > 1$ in (2.23), and Spiriti et al. (2013) study two algorithms for optimising the knot locations in least-squares and penalised splines.

2.3 Change-point detection in panel data

We now consider n univariate data sequences where each sequence consists of T observations collected over time. Then the n-dimensional panel data has a matrix form of the dimension $n \times T$ as follows:

$$\begin{pmatrix} X_{1,1} & X_{1,2} & X_{1,3} & \cdots & X_{1,T} \\ X_{2,1} & X_{2,2} & X_{2,3} & \cdots & X_{2,T} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ X_{n,1} & X_{n,2} & X_{n,3} & \cdots & X_{n,T} \end{pmatrix}. \tag{2.28}$$

This type of multivariate or high-dimensional panel data arises in many different fields including finance, environment, biology, economics, medical sciences and astronomy. If those data sequences in (2.28) experience structural changes at some time points, estimating the number and locations of those change-points is not only often itself of significant interest but also useful for a higher-level representation of the data such as time series clustering or classification. It is indeed a problem of importance in many applications and recent examples include the detection of DNA copy number variants in multiple samples (Zhang et al., 2010), detecting most recent change-point of the events in a telecommunications network recorded for a set of regions (Bardwell et al., 2019), estimation of change-points in average daily river flows recorded in many years (Dette and Gösmann, 2018), detecting change-points in functional magnetic resonance imaging (fMRI) data containing many subjects (Cribben and Yu, 2017; Li et al., 2019), detecting price inflation from UK retail price index data composed of many component indices (Groen et al., 2013).

Change-point detection in panel data introduces many challenges that are not present in the case of a univariate data sequence. Even in the simplest setting with a

single change-point, it may happen that all univariate data sequences have a change-point at different time points or that the change occurs only in a sparse subset of the data sequences. In the following sections, we review how these issues have been dealt with in the literature under various types of structural changes, including changes in mean and/or variance, changes in cross-sectional dependence and changes in polynomial trend. This review covers the existing change-point analyses for both multivariate and high-dimensional panel data where the high-dimensional regime refers to when the dimension n is comparable, or even larger than, the length T of the data stream.

2.3.1 Early works for the case of single change-point

Common change in mean

When the detection of a common change in mean is of interest, the panel data in (2.28) is reformulated as $(\boldsymbol{X}_1, \ldots, \boldsymbol{X}_T)$ where they are independent n-dimensional random vectors sampled from,

$$\boldsymbol{X}_t \sim N_n(\boldsymbol{\mu}_t, \boldsymbol{\Sigma}), \quad 1 \leq t \leq n, \tag{2.29}$$

and $\boldsymbol{\Sigma}$ is non-singular covariance matrix. Many of the early works concern the following hypothesis,

$$H_0 : \boldsymbol{\mu}_1 = \boldsymbol{\mu}_2 = \cdots = \boldsymbol{\mu}_n = \boldsymbol{\mu},$$

against the alternative

$$H_1 : \boldsymbol{\mu}_1 = \cdots = \boldsymbol{\mu}_\eta \neq \boldsymbol{\mu}_{\eta+1} = \cdots = \boldsymbol{\mu}_n, \tag{2.30}$$

where $\boldsymbol{\mu}$ and η are unknown.

Several authors propose Bayesian methods for testing a single mean shift in (2.30). Sen and Srivastava (1973) and Booth and Smith (1982) propose Bayesian statistics

assuming that $\mathbf{\Sigma}$ is the identity matrix and $\mathbf{\Sigma}$ is unknown, respectively. Perreault et al. (2000) apply a Bayesian procedure to detect a single common change in the mean of six hydrological time series. Son and Kim (2005) consider detecting a single change in mean and/or covariance of a sequence of independent multivariate normal vectors.

The likelihood-ratio procedure is popularly used in early studies. Srivastava and Worsley (1986) suggest likelihood ratio tests for the change model in (2.29) and (2.30) with an unknown covariance matrix. Generalising this method, Worsley (1986) studies the detection of a single change in the parameter of the exponential family distribution. Krishnaiah et al. (1987) propose a local likelihood approach for estimating multiple change-points in the mean of multivariate normal distribution. Assuming a single mean shift in (2.29), James et al. (1992) give an asymptotic approximation for the significance level of the likelihood ratio test.

CUSUM is frequently employed for detecting a change in mean vector in the multivariate statistical process control. Woodall and Ncube (1985) propose to use a set of univariate CUSUM procedures for the multivariate case and Healy (1987) discusses detecting a change in mean vector or covariance matrix when the likelihood functions for both before and after the change are known. Crosier (1988) proposes two multivariate CUSUM procedures; the first CUSUM vector is obtained from a univariate series attained by reducing each multivariate observation and the other gives a CUSUM procedure directly from the multivariate observations. Another variant of multivariate CUSUM is introduced in Pignatiello Jr and Runger (1990) and a more complete review of the multivariate CUSUM quality-control can be found in Wierda (1994), Lowry and Montgomery (1995) and Mason et al. (1997). CUSUM statistic is also often used in high-dimensional change-point problem and various aggregating methods are formulated later in Section 2.3.3.

Random change-point

The random change-point model for panel data is first introduced by Joseph (1989). Under the name *multi-path change-point model*, he considers a single change-point in two scenarios as in (2.31); 1) when the change occurs at the same point η in all data sequences (the left matrix) and 2) when the change η_i occurs at random positions in i^{th} data sequence (the right matrix), where i^{th} row corresponds to i^{th} data sequence.

$$\begin{pmatrix} X_{1,1} & \cdots & X_{1,\eta} & | & X_{1,\eta+1} & \cdots & X_{1,T} \\ X_{2,1} & \cdots & X_{2,\eta} & | & X_{1,\eta+1} & \cdots & X_{2,T} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ X_{n,1} & \cdots & X_{n,\eta} & | & X_{1,\eta+1} & \cdots & X_{n,T} \end{pmatrix}, \begin{pmatrix} X_{1,1} & \cdots & X_{1,\eta_1} & | & X_{1,\eta_1+1} & \cdots & X_{1,T} \\ X_{2,1} & \cdots & X_{2,\eta_2} & | & X_{1,\eta_2+1} & \cdots & X_{2,T} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ X_{n,1} & \cdots & X_{n,\eta_n} & | & X_{1,\eta_n+1} & \cdots & X_{n,T} \end{pmatrix}. \tag{2.31}$$

In the case of a common change-point, the distribution of the variables $X_{i,t}$ is assumed to be changed at the change-point η and the bootstrap method is used for approximating the distribution of $\hat{\eta}$, where various methods are explored including maximum likelihood, conditional maximum likelihood, nonparametric and Bayesian in a way of extending the exiting methods for one-dimensional data into multivariate settings. In the case of varying change-points, $\{\eta_i\}_{i=1}^{n}$ is assumed to follow a distribution $G_\eta(t)$ and a few Bayesian techniques are studied with a variety of prior distributions.

We emphasise that these early ideas are established under the assumption that the pre-change data, $\{X_{i,t}\}_{i=1,\ldots,n,t=1,\ldots,\eta_i}$, follow a single distribution f_1 and the post-change data, $\{X_{i,t}\}_{i=1,\ldots,n,t=\eta_i+1,\ldots,T}$, is sampled from another distribution f_2. With this assumption, the multi-path change-point model is improved and applied in Joseph and Wolfson (1992), Joseph and Wolfson (1993), Joseph et al. (1997) and Bélisle et al. (1998). In particular, Joseph et al. (1996) extend the maximum likelihood estimators of Joseph and Wolfson (1993) to correlated observations, where an autoregressive (AR)

process of order p is assumed to the observations from any i^{th} row of (2.28) and the change occurs in its mean and/or autocovariance function. Asgharian and Wolfson (2001) concern the inclusion of covariates in the change-point distribution and study the impact of covariates on the change-point and the parameters. Robinson et al. (2010) extend the maximum likelihood estimation studied in Joseph and Wolfson (1993) and apply it to multiple change-point detection in multi-subject fMRI data.

2.3.2 Multiple change-point detection in multivariate time series

Later works in change-point detection for panel data are developed in a way of considering multiple changes rather than a single change, assuming common change-points rather than random change-points and considering a more complex structure such as dependence across the panel.

Changes in mean

We consider the case when the number and locations of change in mean are of interest. The n-dimensional random vectors, $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_T$, are assumed to have N distinct change-points in mean,

$$0 = \eta_0 < \eta_1 < \eta_2 < \ldots < \eta_N < \eta_{N+1} = T, \tag{2.32}$$

such that

$$\boldsymbol{\mu}_{\eta_\ell+1} = \cdots = \boldsymbol{\mu}_{\eta_{\ell+1}} = \boldsymbol{\mu}^{(\ell)}, \quad \text{for} \quad \ell = 0, \ldots, N, \tag{2.33}$$

$$\boldsymbol{\mu}^{(\ell)} \neq \boldsymbol{\mu}^{(\ell-1)}, \quad \text{for} \quad \ell = 1, \ldots, N, \tag{2.34}$$

where the values of N and $\eta_1, \ldots, \eta_N$ are unknown.

Under this setting, Horváth et al. (1999) propose several test statistics for changes in the mean of multivariate stationary processes. Vert and Bleakley (2010) view this multiple change-point detection as a convex optimisation problem, and find the solution by using a group LARS (Yuan and Lin, 2006). Siegmund et al. (2011) consider the same problem for independent multivariate Gaussian random vectors with an identity covariance matrix. Maboudou-Tchao and Hawkins (2013) propose a maximum likelihood approach for detecting changes in mean and/or covariance of multivariate Gaussian data in the following setting:

$$\eta_\ell + 1 \leq t \leq \eta_{\ell+1}, \quad \boldsymbol{X}_t \sim N_n(\boldsymbol{\mu}^{(\ell)}, \boldsymbol{\Sigma}^{(\ell)}), \tag{2.35}$$

where $\ell = 0, \ldots, N$.

Changes in other features

In a more general context, Lung-Yut-Fong et al. (2011b) consider a nonparametric approach for detecting multiple changes in distribution where no prior knowledge of the distribution is required. To study the same problem, Lung-Yut-Fong et al. (2011a) propose a test statistic by generalising the Mann-Whitney Wilcoxon two-sample test in multivariate settings. Matteson and James (2014) propose nonparametric procedures using both divisive and agglomerative algorithms for hierarchical clustering. In a slightly different setting, Ma and Yau (2016) propose a likelihood-based approach for partitioning an n-dimensional time series into stationary segments and use a pruned dynamic programming for efficient computation. Ombao et al. (2005) also study the segmentation of multivariate nonstationary time series by using a collection of bases named smooth localised complex exponentials in Ombao et al. (2002).

Some authors focus on detecting multiple changes in the cross-covariance structure. Lavielle and Teyssiere (2006) propose an algorithm based on a penalised log-likelihood

and use dynamic programming to compute the optimal path. Aue et al. (2009) propose a nonparametric test based on CUSUM statistic for detecting a structural change in the covariance matrix and extend it to the multiple change-point case via Binary Segmentation. Preuss et al. (2015) suggest a nonparametric procedure for detecting changes in the autocovariance function of a multivariate stationary process by comparing the estimated spectral distribution of different segments. Schröder and Ombao (2019) study the detection of frequency-specific changes in autospectra and coherences for multivariate time series where the procedure is based on a multivariate CUSUM statistics.

Groen et al. (2013) study the detection of multiple changes in regression coefficient by using the average and the maximum of n CUSUM statistics where each component is computed from n-dimensional time series. Kirch et al. (2015) consider detecting change-points in multivariate time series by using the parameters of vector autoregressive (VAR) model as features with application to electroencephalogram (EEG) data. Bardwell et al. (2019) focus on detecting the most recent change-points in panel data,

$$X_{i,t} = f_{i,t} + \varepsilon_{i,t}, \quad i = 1, \ldots, n, \ t = 1, \ldots, T, \tag{2.36}$$

where the signal vectors $\{\boldsymbol{f}_i\}_{i=1}^n$ are assumed to have a form of piecewise-linear function. They first analyse each time series independently through a penalised likelihood approach and post-process to partition n univariate data sequences into a small number of groups that share the most recent change-point.

2.3.3 High-dimensional change-point problem

We consider the high-dimensional settings when both dimension n and the length T can be large and the dimension is comparable, or even larger than, the length of the series. High-dimensional change-point analysis is still in its early stages and has

recently received increasing attention. Many works have considered detecting changes in mean when $\{\boldsymbol{f}_i\}_{i=1}^n$ in (2.36) are modelled as piecewise-constant on which the main focus of our review will be placed. Other related works developed in the context of change-point detection will also be discussed. In what follows, we classify those techniques for high-dimensional time series into few categories and one methodology can be shown in more than one category.

Least-squares criterion

Some authors employ a least-squares criterion to detect a change-point in mean. Bai (2010) is one of the early works in high-dimensional change-point analysis where the consistency of the least-squares estimator of a single change-point is considered. Bhattacharjee et al. (2019) extends the least-square-based approach proposed in Bai (2010) to the case when both temporal and cross-sectional dependences exist.

CUSUM aggregation

The CUSUM procedure has been popularly used in high-dimensional settings. Under the assumption of a single change-point, many different test statistics have been proposed in a way of aggregating CUSUM series, $\mathcal{C}_{1,q,T}^1, \ldots, \mathcal{C}_{1,q,T}^n$, across the panel, where the CUSUM for the i^{th} time series $\{X_{i,t}\}_{t=1}^T$ is defined as:

$$\mathcal{C}_{p,q,r}^i = \sqrt{\frac{r-q}{(r-p+1)(q-p+1)}} \sum_{t=p}^{q} X_{i,t} - \sqrt{\frac{q-p+1}{(r-p+1)(r-q)}} \sum_{t=q+1}^{r} X_{i,t}. \tag{2.37}$$

Note that the CUSUM in (2.37) is the generalised version of the one for univariate time series formulated in (2.16). Zhang et al. (2010) propose a change-point test based on the l_2-aggregated chi-squared statistics derived under the i.i.d. Gaussian assumption. Similarly, Horváth and Hušková (2012) consider a test statistic through a l_2-aggregation

of CUSUM series as:

$$\max_q \left[\frac{q(T-q)}{\sqrt{n}T^2} \sum_{i=1}^{n} \left\{ (\mathcal{C}^i_{1,q,T})^2 - 1 \right\} \right], \tag{2.38}$$

against the alternative hypothesis that the change occurs in all coordinates while Enikeeva and Harchaoui (2013) suggest using a combination of two chi-square type test statistics to enhance the performance in situations where the change occurs only in a subset of coordinates. Cho and Fryzlewicz (2015) consider a l_1-aggregation of the hard-thresholded CUSUM series,

$$\max_q \sum_{i=1}^{n} |\mathcal{C}^i_{1,q,T}| \cdot \mathbb{I}\{|\mathcal{C}^i_{1,q,T}| > \lambda\}, \tag{2.39}$$

as a test statistic for detecting a change in the second-order structure of high-dimensional time series, where λ is a pre-specified threshold. Jirak (2015) suggests the use of the following pointwise maximum (l_∞-aggregation) of the CUSUM statistics as a test statistic:

$$\max_q \max_i \left(\frac{q(T-q)}{T} \right)^{1/2} |\mathcal{C}^i_{1,q,T}|. \tag{2.40}$$

Cho (2016) proposes double CUSUM statistic that is obtained by applying CUSUM transform twice, for each time series first and then again to the sorted CUSUM matrix along the temporal axis. Wang and Samworth (2018) propose a two-stage procedure including the projection of the CUSUM-transformed data and the application of an existing algorithm for univariate change point estimation, where the extension for the multiple change-points is established by borrowing the idea of the Wild Binary Segmentation algorithm (Fryzlewicz, 2014).

Changes in a sparse subset of data sequences

High-dimensional settings consider the case when the number of time series can be very large and it is often too restrictive to assume that all data sequences change at the same locations. As a relaxation of this assumption, some authors focus on cross-sectionally sparse changes.

Without a specific assumption on sparsity, Enikeeva and Harchaoui (2013) employ the scan statistic aimed at improving flexibility in detecting a sparse change and Jirak (2015) studies how to identify the set of coordinates those experience a change. In a slightly different setting, Xie and Siegmund (2013) study the case when a change-point affects only a subset of series, through a mixture procedure based on a generalised likelihood ratio statistics.

Under the sparsity assumptions, Cho and Fryzlewicz (2015) propose sparsified binary segmentation that follows the binary structure for detecting multiple changes but uses the thresholded (or sparsified) CUSUM statistics. Cho (2016) proposes the aggregation of CUSUM statistics through an adaptive partitioning of the panel and Wang and Samworth (2018) apply a sparse singular value decomposition to the CUSUM-transformed data.

Dependences

There are few recent works focusing on both temporal and cross-sectional dependences of high-dimensional time series. Based on a general weak dependence concept, Jirak (2015) studies the asymptotic limit distribution of the coordinate-wise CUSUM statistics. Under both temporal and cross-sectional dependences, Safikhani and Shojaie (2017) study the detection of changes in the coefficients of high-dimensional vector autoregressive (VAR) model. In particular, their method allows the dimension to grow exponentially fast with respect to the length of the series. Bhattacharjee et al.

(2019) propose a least-squares approach for a single change-point where the temporal and cross-sectional dependences are modelled by a moving average error process with infinite order. Li et al. (2019) propose a testing procedure for multiple change-points when both temporal and spatial dependences exist.

Other approaches

Aston and Kirch (2018) study the asymptotic efficiency of the change-point detection test for a single change in mean, that allows us to compare the power of different tests in the high-dimensional settings. Chen and Zhang (2015) propose a graph-based approach for detecting changes in distribution under the assumption that a sequence of n-dimensional observation is independent. Soh and Chandrasekaran (2017) propose a change-point detection method for high-dimensional signals by combining the filtered derivative approach with the convex optimisation. Cribben and Yu (2017) study the detection of changes in network structure in the high-dimensional time series framework. Some other works in covariance change-point detection include Barigozzi et al. (2018) and Wang et al. (2017).

Chapter 3

Smooth-Rough Partitioning of the regression coefficients

3.1 Introduction

In this chapter, we consider random functions $X_i \in L^2[0,1]$ where $i = 1, \ldots, n$ and $[0,1]$ is a compact subset of $\mathbb{R}$. The random functions arise in many contexts e.g. intraday price curves of a financial asset or daily curves of particular matter in the air, and in practice, they are often observed on a grid, rather than continuously. Under the assumption that the repeated realisations of the trajectories are generated by a suitable underlying process, we focus on the random nature of those functions.

The observation of i.i.d. square-integrable random functions $X_i(t) \in L^2[0,1]$ on an equispaced grid $\{t_0, t_1, \ldots, t_T\}$ gives the discretised curves $\{X_i(t_0), X_i(t_1), \ldots, X_i(t_T)\}$ for $i = 1, \ldots, n$ where $t_0 = 0$ and $t_T = 1$. Based on these design points, our objective in this work is to predict the final point $X_i(t_T)$ from the past observations $\{X_i(t_0), \ldots, X_i(t_{T-1})\}$. This is an important applied problem in a variety of fields, including public health, earth sciences, finance and environment, as our data examples in Section 3.5 illustrate. Arguably the simplest statistical framework for expressing

the dependence of $X_i(t_T)$ on $\{X_i(t_0), \ldots, X_i(t_{T-1})\}$ is linearity, and with this in mind, this work focuses on the following model:

$$X_i(t_T) = \mu + \sum_{j=1}^{T} \alpha_j X_i(t_{T-j}) + \varepsilon_i, \quad i = 1, \ldots, n. \tag{3.1}$$

We now discuss its specifics. In our asymptotic considerations, we work with a fixed T, however, in practice, T can be large. For example, two of the datasets in Section 3.5 have T roughly of the order of n which inevitably brings us into a high-dimensional setting and the set of parameters α_j cannot be estimated well by classical approaches. In addition, we often experience a high degree of collinearity between the predictors. As a way of regularising the problem, our proposal in this work is to split the set of parameters $\{\alpha_1, \ldots, \alpha_T\}$ into two sets, $\{\alpha_1, \ldots, \alpha_q\}$ and $\{\alpha_{q+1}, \ldots, \alpha_T\}$, as follows,

$$X_i(t_T) = \mu + \sum_{j=1}^{q} \alpha_j X_i(t_{T-j}) + \sum_{j=q+1}^{T} \alpha_j X_i(t_{T-j}) + \varepsilon_i, \quad i = 1, \ldots, n, \tag{3.2}$$

and assume that the second set, $\{\alpha_{q+1}, \ldots, \alpha_T\}$, is discretised from a smooth curve $\beta(t)$, which gives the model of this chapter:

$$X_i(t_T) = \mu + \sum_{j=1}^{q} \alpha_j X_i(t_{T-j}) + \sum_{j=q+1}^{T} \beta(t_{T-j}) X_i(t_{T-j}) + \varepsilon_i, \quad i = 1, \ldots, n, \tag{3.3}$$

where the final point $X_i(t_T)$ is a scalar response variable, $\{X_i(t_{T-j}),_{j=1,\ldots,T}\} \in \mathbb{R}^T$ represents scalar predictors and ε_i's are iid Gaussian random errors with $E(\varepsilon_i | X_i(t_{T-j}),_{j=1,\ldots,T}) = 0$ and unknown variance σ^2. Since all the dependent and independent variables are obtained from random functions, we assume them to be random. The unknown parameter set contains a constant $\mu \in \mathbb{R}$, real and scalar $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_q)^\top \in \mathbb{R}^q$, real and functional $\beta \in L^2[t_0, t_{T-q-1}]$ and a change-point index parameter q. Throughout the chapter, we will be referring to (3.3) as the Smooth-Rough Partition (SRP) model.

The SRP model assumes that the change-point index q is unknown, and we estimate it from the data via a change-point detection technique. This is possible because we will be assuming that the coefficients α_j are rougher than the coefficients $\beta(t_{T-j})$, i.e. exhibit more variation. One can consider the case of multiple change-points, however in this chapter, we focus on the simplest case when a single change-point exists in smoothness of the regression coefficients.

We now motivate the smooth-rough partitioning idea in more detail. The partitioning of the regression coefficients into two classes of smoothness captures the difference in the relative importance of the observations in predicting the final point $X_i(t_T)$. Constraining the β's to be smooth reflects the relatively lower importance of the more remote observations, whose influence on $X_i(t_T)$ is 'bundled together' by the smoothness restriction in β. By contrast, the unconstrained parameters $\boldsymbol{\alpha}$ are not connected to each other in any (functional) way, so are able to capture any arbitrary linear influence of the near observations on $X_i(t_T)$. The smoothness assumptions on $(\boldsymbol{\alpha}, \beta)$ will be specified in Section 3.3.

The smooth-rough partitioning results in regression estimation that is interpretable in the sense that it automatically separates the effects that can be seen as "long-term" (these are the ones corresponding to the smooth portion of the parameter vector) from those that can be seen as "instantaneous" (these are the ones that correspond to the rough portion of the parameter vector). In other words, the SRP framework can be seen as a "two-scale" approach to linear prediction, where the two scales are defined by both the smoothness and the extent of the regression parameter vector (i.e. the long, smooth portion and the short, rough portion). For example, Figure 3.1 shows that the daily curves of hourly average nitrogen oxides level in Mexico City contain 24 observations each and have similar patterns including two peaks around hours 9 and 21. In the context of the pollution data, it is reasonable to believe that the level of

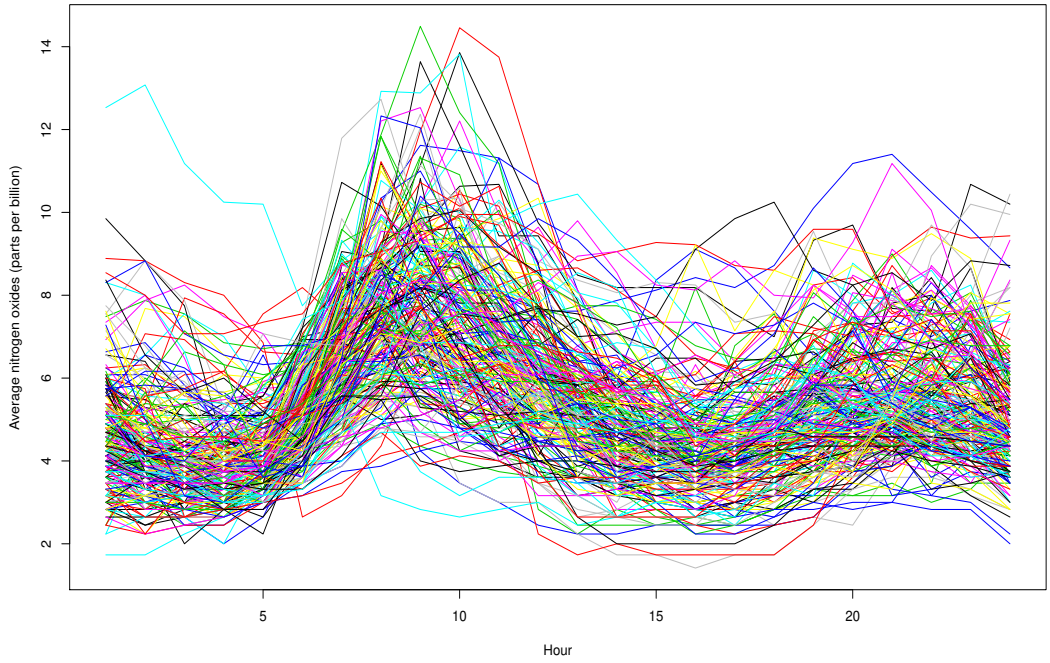


Fig. 3.1 The daily curves of hourly average nitrogen oxides (parts per billion) at the Pedregal station in Mexico City in 2016.

pollution at hour 24 depends both on the overall shape and level of the curve up until the current time i.e. hour 24 (which could be seen as the long-term effect) and the levels immediately preceding the current time (which can be seen as the instantaneous effect). Although we focus on predicting the pollution level at hour 24, if the prediction of a particular hour is of interest, the hours of the curves can be repositioned to put the hour of interest to the final point as a response variable. In Section 3.5.2, we show that those daily curves appear to display both long-term and instantaneous temporal dependences, which are well captured by the SRP model. Besides, in Section 3.5, we demonstrate the usefulness of our two-scale framework in various other real-world datasets e.g. fertility rate data and high-frequency stock volatility series, to which we can attach similar interpretations.

Additionally, the SRP framework can also be useful in the modelling and forecasting of univariate time series, especially those that are believed to be well modelled as AR (autoregressive) processes with large orders. In this case, the smoothing technique of the SRP model would be able to offer both regularisation and interpretability, especially if the time series is believed to possess long memory which will typically be the case if an AR model with a large order is used in the first place. For example, the middle plot of Figure 3.2 shows that the square-rooted monthly sunspot number series may need large-order autoregression (even up to or exceeding order 100), in which case it may be advantageous to use the SRP model over plain AR modelling. In Section 3.5.4, we illustrate that the two-scale framework of the SRP approach is useful in modelling the long memory of a time series.

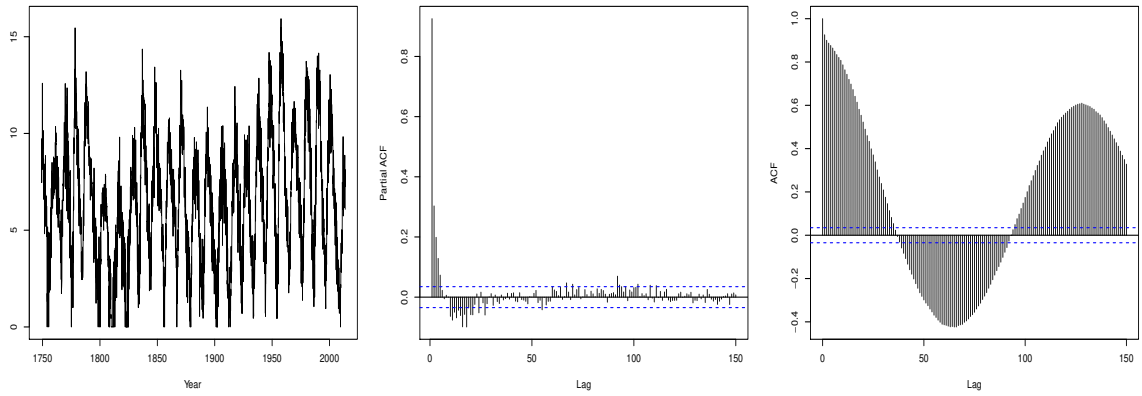


Fig. 3.2 Square-rooted monthly numbers of sunspots from 1749 to 2013 (left), its partial autocorrelation function with maximum lag=150 (middle) and the autocorrelation function with maximum lag=150 (right).

Model (3.3) covers two special cases: 1) in the case of $q = T$, i.e. if we ignore the constrained part, then it has the form of multiple linear regression $X_i(t_T) = \mu + \sum_{j=1}^{T} \alpha_j X_i(t_{T-j}) + \varepsilon_i$ and 2) when $q = 0$, i.e. without the unconstrained part, if the summation is replaced by integration with a large enough T, then it becomes scalar on function regression with $X_i(t_T) = \mu + \int_{t_0}^{t_{T-1}} \beta(t) X_i(t) dt + \varepsilon_i$. Unlike the former, completely unconstrained case, the regularisation in model (3.3) operates in a

way that reduces the model's degrees of freedom. In the examples of Section 3.5, we empirically show that the full model (3.3) exhibits better prediction performance than these two extreme cases. This further justifies our efforts in proposing a methodology for detecting the change-point index q automatically from the data.

We now explain how the SRP model is different from those introduced in Sections 2.1.2-2.1.5 which regularise the functional linear regression coefficient in different ways. In contrast to the 'null subregion' approaches introduced in Section 2.1.2, we do not regularise by finding null subregions of $\beta(t)$ but by imposing different smoothness constraints over different sections of the parameter curve. Capturing two different regimes of smoothness of $\beta(t)$ is done by estimating a change-point which splits the more informative (rough) and the less influential (smooth) regions in $\beta(t)$. In estimating the locations of the influential points by the point of impact approach (Kneip et al., 2016) introduced in Section 2.1.3, they remove the observations adjoining the points of impact, which would be unnecessary in our SRP model as the unrestricted coefficients are grouped into a single region that is the nearest to the time-location of the response variable. While the point of impact approach uses the functional part for estimating the common effect on the entire interval $[0, 1]$, the SRP model uses the smooth functional parameter for a subregion (rather than the entire region) to capture the vanishing memory structure of time series. In Section 3.4, we give examples to show the importance of keeping non-zero (but smooth) part of $\beta(t)$. Some methodologies based on the change-point detection idea are introduced in Section 2.1.4, however neither of these methods use their concept of change-point detection to differentiate between two classes of smoothness, as done by the SRP model. If q in model (3.3) were known, the skeleton of the SRP model is similar to that of partial functional linear regression model in (2.10) in Section 2.1.5. It is worth mentioning that the SRP model studies the case when q is unknown and chooses both independent and dependent

variables from one curve in a time series context. The performance of our technique is compared to some of the methods mentioned above in Sections 3.4 and 3.5.

The remainder of this chapter is organised as follows. Section 3.2 describes the model and the parameter estimation procedure and Section 3.3 presents the relevant theoretical results. The supporting simulation studies are outlined in Section 3.4, with further real-data illustrations in Section 3.5 regarding country fertility data, Mexico city pollution data, stock volatility series and sunspot number data. The technical proofs are in Section 3.6. The SRP methodology is implemented in the R package `srp`.

3.2 Model and its estimation

We work with the discretised curves $\{X_i(t_0), \ldots, X_i(t_T)\}_{i=1,\ldots,n}$ observed from each function $X_i(t)$ on the equispaced $T+1$ discrete points including both endpoints. Since the regression coefficients vary by q, we rewrite model (3.3) as

$$X_i(t_T) = \mu^q + \sum_{j=1}^{q} \alpha_j^q X_i(t_{T-j}) + \sum_{j=q+1}^{T} \beta^q(t_{T-j}) X_i(t_{T-j}) + \varepsilon_i, \quad i = 1, \ldots, n, \tag{3.4}$$

where $1 \leq q \leq T$. The point t_{T-q} is where a sudden smoothness change occurs in the sequence of the regression coefficients, with the coefficients α_j^q being unconstrained in terms of their smoothness and the coefficients $\beta^q(t_{T-j})$ assumed to be a sampled version of a smooth function. Although later the entire function $\beta^q(t)$ is estimated (in a form of function), we only use the points discretised from $\hat{\beta}^q(t)$ and also keep the SRP model with the discrete points $\beta^q(t_{T-j})$ as in (3.4). This is because our model is built on the discretised curves, $\{X_i(t_0), \ldots, X_i(t_T)\}_{i=1,\ldots,n}$, rather than the fully observed curves. The change-point location in (3.4) is the same for all i's. Our expectation is that q is substantially smaller than T and the optimal q is chosen by examining a number of q's over a subset of $\{1, \ldots, T\}$, which we specify in Section 3.2.1. The

reason why T is assumed to be fixed is that if we were to allow $T \to \infty$, then t_T would asymptotically approach t_{T-1} and we could simply predict $X(t_T)$ by $X(t_{T-1})$.

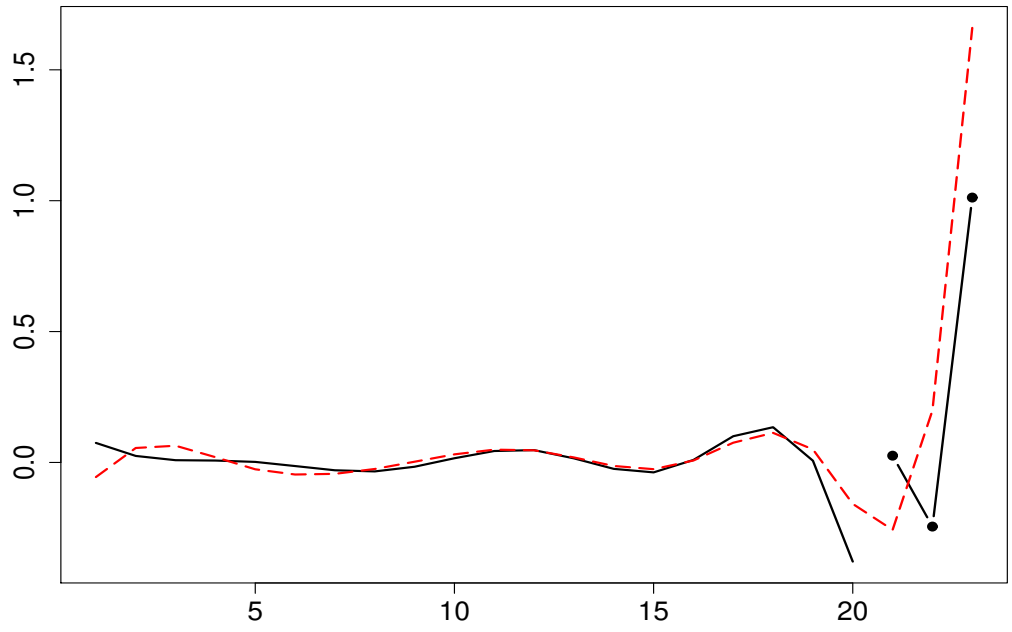


Fig. 3.3 The estimated regression coefficients of the functional linear regression (red dashed) and the SRP model (black) for predicting the average of nitrogen oxides level at hour 24.

We recall the daily curves of hourly average nitrogen oxides in Mexico City shown in Figure 3.1 to motivate the change-point in (3.4). When the final observation recorded at hour 24 is predicted from the past observations indexed 1 to 23, we compare the estimated regression coefficients of the SRP model in (3.4) (i.e. when the model includes both unconstrained and constrained parts in regression parameters) with the estimated functional linear regression coefficient of the model $X_i(t_T) = \mu + \int_{t_0}^{t_{T-1}} \beta(t) X_i(t) dt + \varepsilon_i$ (i.e. when the unconstrained part is ignored in (3.4)). Figure 3.3 shows that those two behave differently especially in the region corresponding to the unconstrained regression coefficients in (3.4). In Section 3.5.2, we show that partitioning the regression parameters into a smooth and a rough regime empirically gives better prediction performance than the functional linear regression model.

The set of unknown parameters in (3.4) can be categorised into two types: 1) change-point t_{T-q} and 2) regression coefficients $(\mu^q, \boldsymbol{\alpha}^q, \beta^q)$. Our interest includes the estimation of the underlying smooth function $\beta(t)$. Broadly speaking, two possible ways exist: 1) estimate $(\hat{\beta}^q(t_0), \ldots, \hat{\beta}^q(t_{T-q-1}))$ and then use interpolation to obtain the functional form of $\hat{\beta}^q(t)$ or 2) obtain the interpolant $\{X(t),\ t \in [t_0, t_{T-q-1}]\}$ and then estimate the function $\hat{\beta}^q(t)$ through basis expansion. In this work, we use the latter approach as it is more popular and the former approach needs a particular penalty to make it feasible if T is close to or exceeding n. Examples of the former can be found in Cardot et al. (2007) and Crambes et al. (2009).

The interpolant $\{X_i(t),\ t \in [t_0, t_{T-q-1}]\}$ is obtained from the discrete observations $(X_i(t_0), \ldots, X_i(t_{T-q-1}))$ using natural cubic splines with knots at $(t_0, \ldots, t_{T-q-1})$. As stated in Crambes et al. (2009), the essential property of natural splines is that for any vector, the unique natural spline interpolant exists and it can be expressed as a B-spline expansion with dimension equal to 'number of knots + 2' (in our case $T - q + 2$) as follows,

$$X_i(t) = \sum_{h=1}^{T-q+2} d_{ih} B_h(t), \quad t \in [t_0, t_{T-q-1}], \tag{3.5}$$

where $B_h(t)$ is a set of basis functions for the normalised B-splines $\{B_h\}_{h=1,\ldots,T-q+2}$. B-splines stands for basis splines as it is used as basis functions for the space of splines. Any spline function can be presented as a unique linear combination of B-splines, where a spline function is a piecewise polynomial function.

As stated in Section 2.1.1, dimension reduction is necessary for the estimation of $\beta(t)$. In what follows, we use B-splines. Cardot et al. (2003) argue that spline estimators should be preferred to the functional PC approach when $X(t)$ is rough and the functional coefficient is smooth, which is the case we are interested in. Moreover, a spline estimator is not directly affected by the estimation of the eigenstructure of the covariance operator of $X(t)$.

Let $\mathcal{S}$ be the space of splines defined on $[t_0, t_{T-q-1}]$ with degree s and $k-1$ equispaced interior knots where $L = k + s$ denotes the dimension of $\mathcal{S}$. Then one can derive a set of basis functions from the normalised B-splines $\{B_l\}_{l=1,\dots,L}$ to approximate $\beta^q(t)$ as

$$\beta^q(t) \approx \sum_{l=1}^{L} b_l^q B_l(t), \quad t \in [t_0, t_{T-q-1}], \tag{3.6}$$

where b_l represents the corresponding coefficient. For each t_{T-q}, the set of the regression parameters simplifies to $\boldsymbol{\delta}^q = (\mu^q, \boldsymbol{\alpha}^q, b_1^q, \dots, b_L^q)^\top \in \mathbb{R}^{1+q+L}$ where $\boldsymbol{\alpha}^q = (\alpha_1^q, \dots, \alpha_q^q)^\top$. The choice of L is considered in Section 3.2.2.

3.2.1 Joint estimation procedure for parameters

We suggest a one-stage estimation procedure for the change-point and the regression parameters. Since the parameter q represents the number of scalar parameters, under fixed L, q itself determines the dimension of the model. Thus, using the well-known criterion of Schwarz (1978), we estimate q by minimising

$$SIC(q) = n \cdot \log M(q) + (q + L + 1) \cdot \log n, \tag{3.7}$$

where

$$M(q) = \frac{1}{n} \sum_{i=1}^{n} \left\{ X_i(t_T) - \hat{\mu}^q - \sum_{j=1}^{q} \hat{\alpha}_j^q X_i(t_{T-j}) - \sum_{j=q+1}^{T} \hat{\beta}^q(t_{T-j}) X_i(t_{T-j}) \right\}^2, \tag{3.8}$$

and $(\hat{\mu}^q, \hat{\alpha}_j^q, \hat{\beta}^q(t_{T-j}))$ are repeatedly estimated for each q by minimising the following sum of squared errors with appropriate penalisations,

$$(\hat{\boldsymbol{\alpha}}^q, \hat{\beta}^q(t)) = \underset{\boldsymbol{\alpha}^q, \beta^q(t)}{\arg\min} \left[\frac{1}{n} \sum_{i=1}^{n} \left\{ \tilde{X}_i(t_T) - \sum_{j=1}^{q} \alpha_j^q \tilde{X}_i(t_{T-j}) - \int_{t_0}^{t_{T-q-1}} \beta^q(t) \tilde{X}_i(t) dt \right\}^2 + \lambda_1 \boldsymbol{\delta}_0^{q\top} \boldsymbol{\delta}_0^q + \lambda_2 \int_{t_0}^{t_{T-q-1}} \left\{ \beta^{q(m)}(t) \right\}^2 dt \right], \tag{3.9}$$

$$\hat{\mu}^q = \bar{X}(t_T) - \sum_{j=1}^{q} \hat{\alpha}_j^q \bar{X}(t_{T-j}) - \sum_{j=q+1}^{T} \hat{\beta}^q(t_{T-j}) \bar{X}(t_{T-j}),$$

where $\boldsymbol{\delta}_0^q = (\boldsymbol{\alpha}^q, b_1^q, \ldots, b_L^q)^\top \in \mathbb{R}^{q+L}$, $\tilde{X}_i(t_{T-j})$ and $\tilde{X}_i(t)$ are demeaned predictors, $\bar{X}(t_{T-j}) = \frac{1}{n} \sum_{i=1}^{n} X_i(t_{T-j})$ and $\beta^{q(m)}(t)$ is the m^{th} derivative of $\beta^q(t)$ with the positive integer m satisfying $m < s$ where s denotes the degree of space $\mathcal{S}$. We note that $M(q)$ in (3.8) comes from the Gaussian error assumption in model (3.3) on which SIC can be written in terms of the residual sum of squares (RSS) as $SIC = n \cdot \log(\text{RSS}/n) + (q + L + 1) \cdot \log n$. Importantly, in Section 3.6, it will be shown that the $\log n$ in the penalty term of SIC (which is larger than that of AIC (Akaike, 1974)) plays an important role in achieving the consistency of the estimated change-point index parameter q, and this justifies the usage of SIC in estimating q.

The penalty terms in (3.9) contain two tuning parameters: λ_1 controls a ridge-type penalty and λ_2 governs the smoothness of the estimated $\hat{\beta}^q(t)$. In practice, only the initial values of λ_1 and λ_2 need to be specified by the user and the optimal values are selected automatically via a cross-validation-type criterion described in Section 3.2.2. If q were known, our task would be to estimate the regression parameters $(\mu^q, \boldsymbol{\alpha}^q, \beta^q)$. However, we assume that q is not known and estimate the parameters $(q, \mu^q, \boldsymbol{\alpha}^q, \beta^q)$ jointly. We preserve the original time scale of $\beta^q(t)$ instead of rescaling it to $[0, 1]$ so that we can place $\hat{\boldsymbol{\alpha}}^{\hat{q}}$ and $\hat{\beta}^{\hat{q}}(t)$ on the same time scale.

We present a procedure for the estimation of regression parameters in (3.9). Under a fixed q, one of those components in (3.9) can be approximated by the basis expansions described in (3.5) and (3.6) as follows:

$$\int_{t_0}^{t_{T-q-1}} \beta^q(t)\tilde{X}_i(t)dt \approx \sum_{h=1}^{T-q+2}\sum_{l=1}^{L} d_{ih}\Big\{\int_{t_0}^{t_{T-q-1}} B_h(t)B_l(t)dt\Big\}b_l^q = \boldsymbol{d}_i^\top \mathbf{J}^q \boldsymbol{b}^q, \qquad (3.10)$$

where $\boldsymbol{d}_i = (d_{i,1}, ..., d_{i,T-q+2})^\top$, $\boldsymbol{b}^q = (b_1^q, ..., b_L^q)^\top$ and $\mathbf{J}^q$ is a matrix of the dimension $(T-q+2)\times L$ with its $(h,l)^{th}$ element, $\mathbf{J}_{hl}^q = \int_{t_0}^{t_{T-q-1}} B_h(t)B_l(t)dt$. The two penalty terms in (3.9) can be reconstructed as $\boldsymbol{\delta}_0^{q\top}\mathbf{R}^q\boldsymbol{\delta}_0^q = \lambda_1\boldsymbol{\delta}_0^{q\top}\boldsymbol{\delta}_0^q + \lambda_2\int_{t_0}^{t_{T-q-1}}\{\beta^{q(m)}(t)\}^2dt$ where

$$\mathbf{R}^q_{(q+L)\times(q+L)} = \begin{bmatrix} \lambda_1\mathbf{I}_q & \mathbf{0} \\ \mathbf{0} & \lambda_1\mathbf{I}_L + \lambda_2\mathbf{R}_0^q \end{bmatrix}, \qquad (3.11)$$

and $\mathbf{R}_0^q$ is a $L\times L$ matrix with its $(h,l)^{th}$ element, $\mathbf{R}_{0,hl}^q = \int_{t_0}^{t_{T-q-1}}\{B_h^{(m)}(t)\}\{B_l^{(m)}(t)\}dt$. The penalised minimisation in (3.9) can be simplified as

$$PENSSE_{\lambda_1,\lambda_2}[\boldsymbol{\delta}_0^q = (\boldsymbol{\alpha}^q, \boldsymbol{b}^q)^\top] = \parallel \tilde{\boldsymbol{X}}(t_T) - (\tilde{\mathbf{X}}\boldsymbol{\alpha}^q + \boldsymbol{D}\boldsymbol{J}^q\boldsymbol{b}^q) \parallel^2 + \boldsymbol{\delta}_0^{q\top}\mathbf{R}^q\boldsymbol{\delta}_0^q, \quad (3.12)$$

where $\tilde{\mathbf{X}}_{n\times q} = (\tilde{\boldsymbol{X}}(t_{T-1}), ..., \tilde{\boldsymbol{X}}(t_{T-q}))$ and $\boldsymbol{D}_{n\times(T-q+2)} = (\boldsymbol{d}_1^\top, ..., \boldsymbol{d}_n^\top)$. Given some tuning parameters λ_1 and λ_2, the minimiser $\hat{\boldsymbol{\delta}}_0^q$ can be attained as a closed form of

$$\hat{\boldsymbol{\delta}}_0^q = (\boldsymbol{A}^{q\top}\boldsymbol{A}^q + \boldsymbol{R}^q)^{-1}\boldsymbol{A}^{q\top}\tilde{\boldsymbol{X}}(t_T) \qquad (3.13)$$

where $\boldsymbol{A}^q = [\tilde{\mathbf{X}} \quad \boldsymbol{D}\boldsymbol{J}^q]$ is the design matrix.

We now consider why the minimisation of the SIC penalty in (3.7) is particularly useful in estimating q. First, let $q_0, \boldsymbol{\alpha}_0, \beta_0$ denote the true values of the parameters $q, \boldsymbol{\alpha}, \beta$, respectively. The left plot in Figure 3.4 shows that as a function of q, $M(q)$ typically decreases sharply as $q \uparrow q_0$, and becomes relatively flat (as $n\to\infty$) for $q \geq q_0$.

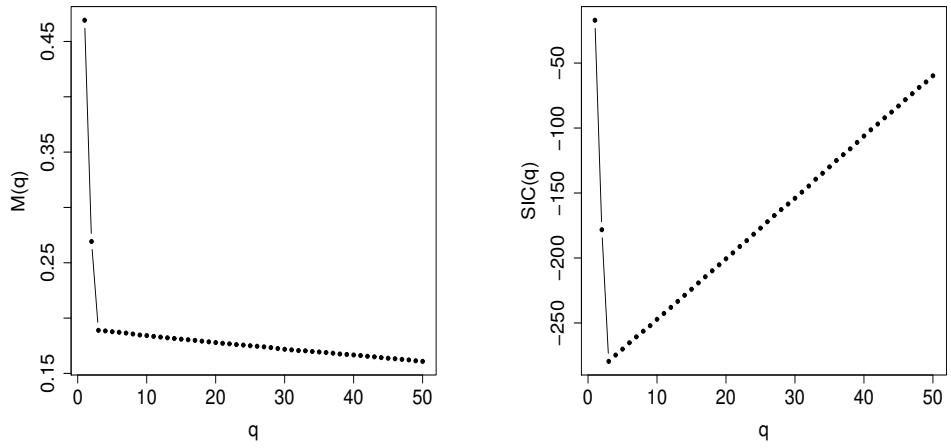


Fig. 3.4 Mean of $\{M(q)\}_{1\leq q\leq 50}$ (left) and $\{SIC(q)\}_{1\leq q\leq 50}$ (right) defined in (3.8) and (3.7) (respectively) over 100 simulation runs for Case 2 defined in Section 3.4, in which case q_0=3.

For $q > q_0$, as q is larger than the true one, the smooth function $\beta_0(t)$ on the interval $[t_{T-q}, t_{T-q_0-1}]$ is inevitably estimated by the scalar estimators $(\hat{\alpha}_q, \ldots, \hat{\alpha}_{q_0+1})$, in which case the scalar estimators $(\hat{\alpha}_q, \ldots, \hat{\alpha}_{q_0+1})$ are obtained in a relatively flexible way in that the smoothness is unrestricted, therefore the fit is typically good which causes the flat shape of $M(q)$ when $q > q_0$. Conversely, when $q < q_0$, some of the unrestricted parameters, $(\alpha_{0,q_0}, \ldots, \alpha_{0,q+1})$, are estimated by $(\hat{\beta}^q(t_{T-q_0}), \ldots, \hat{\beta}^q(t_{T-q-1}))$ under a smoothness restriction, which typically causes $M(q)$ to be away from its minimum for $q < q_0$. The right plot in Figure 3.4 shows that the SIC penalty "lifts" the flat part of $M(q)$ and enables us to estimate the q parameter close to its true value. This is shown theoretically in Section 3.3 and numerically in Sections 3.4 and 3.5.

When finding the optimal q in (3.7), although q can in principle be large enough up to $q = T$, we recommend examining $1 \leq q \leq \bar{q}$, where $\bar{q}$ is substantially smaller than T. In the examples considered in Sections 3.4 and 3.5, we take $\bar{q} = \min(\lceil T \times 0.1 \rceil, 30)$. Based on our empirical experience, when q is large, there is the possibility that the optimisation of the two tuning parameters, λ_1 and λ_2 in (3.9), becomes unstable in

that it becomes highly dependent on the selection of their initial values. In addition, examining the entire range $1 \leq q \leq T$ can make the algorithm unnecessarily slow especially when both T and n are large. In practice, even if we do not restrict q to be small as stated above, the minimiser $\hat{q}$ of $SIC(q)$ in (3.7), if computed successfully despite the potential stability issues, is typically obtained to be substantially smaller than T.

3.2.2 Selection of the tuning parameters

To select the tuning parameters, we use the `magic` function from the R package `mgcv` (Wood (2006)). The `mgcv` includes various regression models such as GAM or the generalised ridge regression. The `magic` function is useful in that it is able to optimise over more than one penalty parameters (λ_1 and λ_2 in our case) by minimising GCV based on Newton's method where GCV function is as follows:

$$\mathrm{GCV}(\lambda_1, \lambda_2) = \frac{n \parallel (\mathbf{I} - \boldsymbol{A}^*(\lambda_1, \lambda_2))\tilde{\boldsymbol{X}}(t_T) \parallel^2}{[\mathrm{tr}(\mathbf{I} - \boldsymbol{A}^*(\lambda_1, \lambda_2))]^2}, \tag{3.14}$$

where $\boldsymbol{A}^*(\lambda_1, \lambda_2) = \boldsymbol{A}^q(\boldsymbol{A}^{q\top}\boldsymbol{A}^q + \boldsymbol{R}^q)^{-1}\boldsymbol{A}^{q\top}$ and $(\mathbf{R}^q, \boldsymbol{A}^q)$ can be found in (3.11) and (3.13), respectively. The practical use of the `magic` function in our setting is as follows:

`magic( y, X, sp, S, off ).`

Note that `y` is the response vector, `X` is the design matrix, `sp` is the starting values for optimising penalty parameters, `S` is a list of penalty matrices and `off` is an array indicating the locations of the first parameter penalised by the corresponding penalty matrices in `S`. In our case, the penalty matrix $\mathbf{R}^q$ in (3.11) can be represented as the sum of two matrices where each contains the corresponding penalty parameters λ_1 and

λ_2 as follows,

$$\mathbf{R}^q_{(q+L)\times(q+L)} = \begin{bmatrix} \lambda_1\mathbf{I}_q & \mathbf{0} \\ \mathbf{0} & \lambda_1\mathbf{I}_L + \lambda_2\mathbf{R}^q_0 \end{bmatrix} = \lambda_1\mathbf{I}_{q+L} + \lambda_2 \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}^q_0 \end{bmatrix},$$

therefore for a certain q, the `magic` function has the form of

`magic(` y=$\tilde{\boldsymbol{X}}(t_T)$, X=$\boldsymbol{A}^q$, sp=$c(1,1)$, S=list($\mathbf{I}_{q+L}$, $\mathbf{R}^q_0$), off=$c(1, q+1)$).

The results give the optimal penalty parameters, $\hat{\lambda}_1$ and $\hat{\lambda}_2$, and also the estimators $(\hat{\boldsymbol{\alpha}}^q, \hat{\beta}^q(t))$ in (3.9) under a certain q.

Regarding the dimension of β^q, we typically set L to be large but substantially smaller than $T - q$. As mentioned in Ruppert (2002), the number of basis functions tends not to play an important role in functional linear regression with a roughness penalty, if we choose it to be large enough to prevent undersmoothing. Following the rule of thumb from Ruppert (2002), we use $L = 35$ in Sections 3.4 and 3.5, except in cases in which $T < 40$, when we use $L = 9$.

3.3 Theoretical results

In this section, we assume that the SRP model in (3.4) is correct and explore the asymptotic behaviour of $\hat{q}$, the estimator of the change-point index q_0. There is a one-to-one correspondence between q and t_{T-q}, so we will be interchangeably considering $\hat{q}$ and $t_{T-\hat{q}}$. We denote the true values of scalars $\boldsymbol{\alpha}$ and function β by $(\boldsymbol{\alpha}_0, \beta_0)$ and assume the following conditions.

Assumption 3.1 *$\beta_0(t)$ is continuous on $t \in [t_0, t_{T-q_0-1}]$ and $\boldsymbol{\alpha}_0$ is composed of the finite number of scalars $\boldsymbol{\alpha}_0 = (\alpha_{0,1}, \ldots, \alpha_{0,q_0})^\top$ on $t \in [t_{T-q_0}, t_{T-1}]$.*

Assumption 3.2 *The true change-point $t_{T-q_0} \in (t_0, t_{T-1}]$ is where the change of smoothness occurs in the sequence of true regression parameters. When $q_0 > 1$, taking*

q_1 such that $1 \leq q_1 < q_0$, for any $q \in [q_1, q_0)$, there exist $\delta_1, \delta_2, \delta_3 > 0$ and $c_1, c_2, c_3 > 0$ such that $(a)\, \inf_{1\leq j\leq q} |\alpha_{0,j} - \hat{\alpha}_j^q| > \delta_1$, $\sup_{1\leq j\leq q} |\alpha_{0,j} - \hat{\alpha}_j^q| \leq c_1$ $(b)\, \inf_{q_0<j\leq T} |\beta_0(t_{T-j}) - \hat{\beta}^q(t_{T-j})| > \delta_2$, $\sup_{q_0<j\leq T} |\beta_0(t_{T-j}) - \hat{\beta}^q(t_{T-j})| \leq c_2$ *and* $(c)\, \inf_{q<j\leq q_0} |\alpha_{0,j} - \hat{\beta}^q(t_{T-j})| > \delta_3$, $\sup_{q<j\leq q_0} |\alpha_{0,j} - \hat{\beta}^q(t_{T-j})| \leq c_3$.

Assumption 3.3 *Taking q_2 such that $1 \leq q_0 < q_2 < T$,*

$$
\begin{aligned}
(a) \quad & \sup_{q_0\leq q\leq q_2} \frac{1}{n}\sum_{i=1}^{n}\left[\sum_{1\leq j\leq q_0}(\alpha_{0,j} - \hat{\alpha}_j^q)\tilde{X}_i(t_{T-j})\right]^2 = O_p(n^{-1}),\\
(b) \quad & \sup_{q_0\leq q\leq q_2} \frac{1}{n}\sum_{i=1}^{n}\left[\sum_{q<j\leq T}(\beta_0(t_{T-j}) - \hat{\beta}^q(t_{T-j}))\tilde{X}_i(t_{T-j})\right]^2 = O_p(n^{-1}),\\
(c) \quad & \sup_{q_0<q\leq q_2} \frac{1}{n}\sum_{i=1}^{n}\left[\sum_{q_0<j\leq q}(\beta_0(t_{T-j}) - \hat{\alpha}_j^q)\tilde{X}_i(t_{T-j})\right]^2 = O_p(n^{-1}).
\end{aligned}
$$

Assumption 3.4 *When q_2 is as in Assumption 3.3,*

$$
\begin{aligned}
(a) \quad & \sup_{q_0\leq q\leq q_2} \left|\frac{1}{n}\sum_{i=1}^{n}(\varepsilon_i - \bar{\varepsilon})\sum_{1\leq j\leq q_0}(\alpha_{0,j} - \hat{\alpha}_j^q)\tilde{X}_i(t_{T-j})\right| = O_p(n^{-1}),\\
(b) \quad & \sup_{q_0\leq q\leq q_2} \left|\frac{1}{n}\sum_{i=1}^{n}(\varepsilon_i - \bar{\varepsilon})\sum_{q<j\leq T}(\beta_0(t_{T-j}) - \hat{\beta}^q(t_{T-j}))\tilde{X}_i(t_{T-j})\right| = O_p(n^{-1}),\\
(c) \quad & \sup_{q_0<q\leq q_2} \left|\frac{1}{n}\sum_{i=1}^{n}(\varepsilon_i - \bar{\varepsilon})\sum_{q_0<j\leq q}(\beta_0(t_{T-j}) - \hat{\alpha}_j^q)\tilde{X}_i(t_{T-j})\right| = O_p(n^{-1}).
\end{aligned}
$$

Assumption 3.5 *The independent and identically distributed errors ε_i are independent of the predictors. We further assume $E(\boldsymbol{X}^\top \boldsymbol{X}) + E(\varepsilon^2) < \infty$ with $E(\varepsilon) = 0$, where $\boldsymbol{X}_{n\times T} = (X(t_0), X(t_1), \ldots, X(t_{T-1}))$.*

Assumption 3.6 *Writing the singular value decomposition of the covariance matrix of $\boldsymbol{X}$ as $K_{(k_1,k_2)} = cov(X(t_{k_1}), X(t_{k_2})) = \sum_{j=1}^{T} v_j \boldsymbol{\psi}_j \boldsymbol{\psi}_j^\top$ where $v_1 \geq v_2 \cdots > 0$ are eigenvalues, and $\boldsymbol{\psi}_1, \boldsymbol{\psi}_2, \ldots$ are the corresponding eigenvectors, we assume that the eigenvalues decay sufficiently fast so that the condition $\sum_{j=1}^{T} v_j^{1/2} \left\|\boldsymbol{\psi}_j\right\|_\infty < \infty$ holds.*

Assumption 3.2 quantifies the non-convergences occurring when $q < q_0$ as mentioned in the discussion of the shape of the function $M(q)$ in Section 3.2.1. It can be seen that Assumption 3.2 is somewhat strong as it states that the non-convergences occur in all regions of the estimated regression coefficients. Based on our empirical experience, among three non-convergences stated in Assumption 3.2, the last component (i.e. $|\alpha_{0,j} - \hat{\beta}^q(t_{T-j})|$) generally has the most significant effect on the non-convergence of $M(q)$. However, this is not always the case as both of the smooth and the rough regression parameters are estimated at once by minimising (3.9) in a way of affecting each other, which implies that the non-convergence caused by estimating the rough part ($\alpha_{0,j}$) through the smooth estimator ($\hat{\beta}^q(t_{T-j})$) possibly induces the non-convergence of the estimators in other regions such as $\alpha_{0,j}$ estimated by $\hat{\alpha}_j^q$ and/or $\beta_0(t_{T-j})$ estimated by $\hat{\beta}^q(t_{T-j})$.

In contrast to Assumption 3.2, Assumptions 3.3 and 3.4 list the converging components of $M(q)$ when $q \geq q_0$, where those can be considered as a discrete version of the following assumptions made on the estimated regression coefficients in Hall and Hooker (2016):

$$\sup_{\theta_1 \leq \theta \leq \theta_2} \frac{1}{n} \sum_{i=1}^{n} \left[\int_0^{\theta} (\hat{\beta}(t) - \beta_0(t))(X_i(t) - \bar{X}(t))dt \right]^2 = O_p(n^{-1}), \tag{3.15}$$

$$\sup_{\theta_1 \leq \theta \leq \theta_2} \left| \frac{1}{n} \sum_{i=1}^{n} (\varepsilon_i - \bar{\varepsilon}) \int_0^{\theta} (\hat{\beta}(t) - \beta_0(t))(X_i(t) - \bar{X}(t))dt \right| = O_p(n^{-1}), \tag{3.16}$$

where θ_1 and θ_2 satisfy $0 \leq \theta_1 < \theta_0 < \theta_2 \leq 1$ and θ_0 is the true truncation point. As has been noted by many researchers (see e.g. Pumo (1998), Cardot et al. (2003), Cardot et al. (2007), Crambes et al. (2009), Descary and Panaretos (2019)), discretisation of the curves X has no effect on the convergence rate of the regression parameter if the number of discretisation point is sufficiently large. Specifically, Cardot et al. (2003) find that when $\hat{\beta}(t)$ is spanned by usual splines (e.g. B-splines), the equal rate

of convergence is guaranteed under the condition that the longest distance between grid points, $\max_i|t_{i+1} - t_i|$, converges to zero fast enough compared to the number of knots chosen for the space of B-splines when the number of observations n goes to infinity. Hall and Hooker (2016) investigate the case when the set of eigenfunctions of the covariance operator of $X(t)$ are used, by which the B-spline expansion employed in this chapter can be replaced. They mention that the methods used by Cai and Hall (2006) can give the rate of convergence of $\beta^q(t)$ in $(n^{-1/2}, n^0)$ for Assumption 3.3-(b) under appropriate smoothness conditions for $\beta(t)$, $X(t)$ and the covariance function, measured by the spacing of the eigenvalues in a full functional setting (that is, when $q = 0$ in our case). Similarly, Crambes et al. (2009) derive the rate of convergence for the general spline classes which is comparable to that of Cai and Hall (2006), under the usual smoothness assumptions on $\beta(t)$ and $X(t)$ defined by the continuity of its derivatives. The methods used in Crambes et al. (2009) can give the rate in Assumption 3.3-(b) under appropriate smoothness conditions for $\beta(t)$ and $X(t)$ in a full functional setting. Since our model contains scalar covariates and has the ridge type penalty in (3.9), we postulate the same or slightly slower rates, which are also supported by our numerical experience. Assumptions 3.5 and 3.6 are used for establishing Lemma 3.1 in Section 3.6, that is related to the corresponding components in Assumption 3.4 when $q < q_0$ as in Assumption 3.2.

We are now ready to state our main result.

Theorem 3.1 *If $\hat{q}$ is any value of q which minimises (3.7) on the interval $[q_1, q_2]$ when q_1 and q_2 are chosen to satisfy $1 \leq q_1 < q_0 < q_2 < T$, then under Assumptions 3.1–3.6, we have $P(\hat{q} = q_0) \to 1$ as $n \to \infty$.*

The result of Theorem 3.1 agrees with the numerical evidence of the increased closeness of $\hat{q}$ to q_0 as n increases that is illustrated in Figure 3.7. Technical proof is available in Section 3.6.

3.4 Simulations

In this section, we evaluate the finite-sample performance of our approach. We expect the performance of our method to vary depending on the size of change between $\beta_0(t)$ and $\boldsymbol{\alpha}_0^\top$ (i.e. $|\alpha_{0,q_0} - \beta_0(t_{T-q_0-1})|$) and on the degree of fluctuations in the $\boldsymbol{\alpha}_0^\top$ coefficients relative to the smoothness of $\beta_0(t)$. The true signals are available from the R package `srp`.

Based on the model (3.4), we consider the following four parametric cases, Case 1: $\mu_0 = 0.0180, \boldsymbol{\alpha}_0 = (0.4, 0.2, 0.1)^\top$, Case 2: $\mu_0 = -0.0836, \boldsymbol{\alpha}_0 = (0.6, -0.5, 0.4)^\top$, Case 3: $\mu_0 = -0.0239, \boldsymbol{\alpha}_0 = (0.4, 0.2, 0.1)^\top$ and Case 4: $\mu_0 = -0.0742, \boldsymbol{\alpha}_0 = (0.4, -0.2, 0.1)^\top$, to investigate how the performance of change-point detection is affected by the degree of changes in the regression parameters. The true change-point index parameter is $q_0 = 3$ for all cases as shown in Figure 3.5 where $\beta_0(t)$ and $\boldsymbol{\alpha}_0^\top$ are plotted on a different scale. In the data generating process based on the model (3.4), we use the Gaussian noise ε_i with the signal-to-noise ratio, defined as $\sigma^2_{\mathbf{X}}/\sigma^2$, equal to 4 where $\sigma^2_{\mathbf{X}} = \text{var}(X(t_T) - \varepsilon)$ and σ^2 is the error variance. In Cases 1 and 3, $\boldsymbol{\alpha}_0$ shows less fluctuation than in Cases 2 and 4. The size $|\alpha_{0,3} - \beta_0(t_{T-4})|$ of the change-point is approximately 0.4 in Case 2 and approximately 0.1 in the remaining three cases. Case 3 is similar to Case 1 except that its $\beta_0(t) = b_0 + b_1 t$ is linear. We simulate $n = 300$ independent copies of each process, in which the length of the sample is $T + 1 = 360$ (see formula (3.4)).

In each of 100 Monte Carlo runs, we split $n = 300$ observations into training and test sets of sizes $n_1 = 150$ and $n_2 = 150$, respectively. The training sample is used to obtain $\hat{q}$ and $(\hat{\boldsymbol{\alpha}}, \hat{\beta})$ by minimising (3.7) and (3.9), respectively. The accuracy of the regression parameter estimators can be evaluated by comparing $(\hat{\boldsymbol{\alpha}}^q, \hat{\beta}^q(t))$ and $(\boldsymbol{\alpha}_0, \beta_0(t))$; however, if the change-point is incorrectly estimated, i.e. $\hat{q} \neq q_0$, the length of the vector $\hat{\boldsymbol{\alpha}}^q$ is not matched with that of $\boldsymbol{\alpha}_0$ and

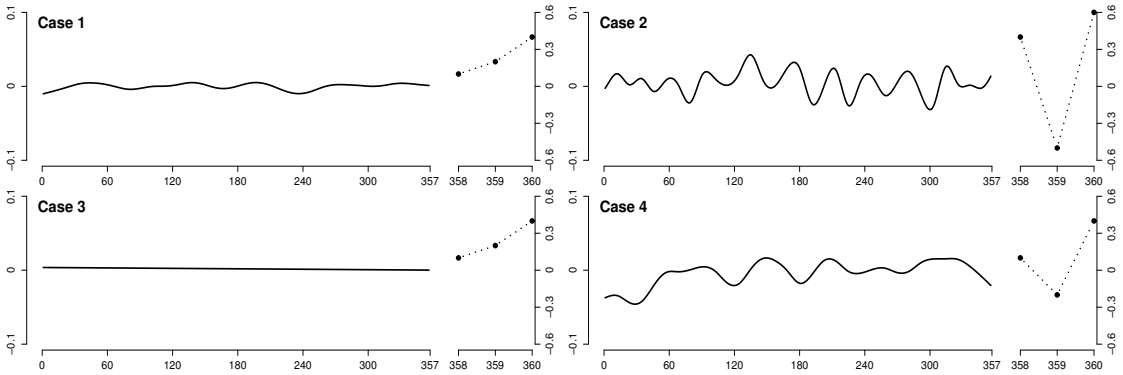


Fig. 3.5 True regression parameters of Cases 1-4 with different scale for each $\beta_0(t)$ (solid line) and $\boldsymbol{\alpha}_0^\top$ (dots).

neither is $\hat{\beta}^q(t)$. To circumvent this, we discretise $\hat{\beta}^q(t)$ and $\beta_0(t)$ and define $\hat{\boldsymbol{\gamma}}_{\hat{q}}$ and $\boldsymbol{\gamma}_0$ of dimension $T \times 1$ as $\hat{\boldsymbol{\gamma}}_{\hat{q}} = \left(\hat{\alpha}_1^{\hat{q}}, \ldots, \hat{\alpha}_{\hat{q}}^{\hat{q}}, \hat{\beta}^{\hat{q}}(t_0), \ldots, \hat{\beta}^{\hat{q}}(t_{T-\hat{q}-1})\right)^\top$ and $\boldsymbol{\gamma}_0 = \left(\alpha_{0,1}, \ldots, \alpha_{0,q_0}, \beta_0(t_0), \ldots, \beta_0(t_{T-q_0-1})\right)^\top$, which enables us to use the following sum-of-squared-errors (SSE) criterion:

$$\text{SSE} = \left[\hat{\boldsymbol{\gamma}}_{\hat{q}} - \boldsymbol{\gamma}_0\right]^\top \left[\hat{\boldsymbol{\gamma}}_{\hat{q}} - \boldsymbol{\gamma}_0\right]. \tag{3.17}$$

The prediction performance is examined in the test sample by computing the mean-square prediction error (MSPE),

$$\text{MSPE} = \frac{1}{n_2} \sum_{i=1}^{n_2} \{X_i(t_T) - \hat{X}_i(t_T)\}^2, \tag{3.18}$$

where $\hat{X}_i(t_T)$ is the prediction using the estimated parameters $(\hat{q}, \hat{\mu}^{\hat{q}}, \hat{\boldsymbol{\alpha}}^{\hat{q}}, \hat{\beta}^{\hat{q}}(t))$.

3.4.1 Competing methods

We compare the performance of our approach to the following existing methodologies: multiple linear regression (**MLR**), ridge regression (**RIDGE**), functional linear regression with penalised B-splines (**FLR**, Cardot et al. (2003)), interpretable functional

linear regression (**FLiRTI**, James et al. (2009)), most-predictive design points approach (**MPDP**, Ferraty et al. (2010)) and functional nonparametric regression (**NP**, Ferraty and Vieu (2002)). We also compare our proposal ($\mathbf{SRP}_{\mathcal{C}}$) with its simplified version named $\mathbf{SRP}_{\mathcal{L}}$, which follows the form of $\mathbf{SRP}_{\mathcal{C}}$ except that $\beta_0(t)$ is estimated as a linear function. The corresponding objective functions for the parametric methods are as follows:

$$\mathbf{MLR}: \hat{\boldsymbol{\alpha}}^{\hat{q}_1} = \underset{\boldsymbol{\alpha}^{\hat{q}_1}}{\arg\min} \frac{1}{n_1} \sum_{i=1}^{n_1} \left\{ \tilde{X}_i(t_T) - \sum_{j=1}^{\hat{q}_1} \alpha_j^{\hat{q}_1} \tilde{X}_i(t_{T-j}) \right\}^2,$$

$$\mathbf{FLR}: \hat{\beta}(t) = \underset{\beta(t)}{\arg\min} \frac{1}{n_1} \sum_{i=1}^{n_1} \left\{ \tilde{X}_i(t_T) - \int_{t_0}^{t_{T-1}} \beta(t) \tilde{X}_i(t) dt \right\}^2 + \lambda \int_{t_0}^{t_{T-1}} \left\{ \beta^{(m)}(t) \right\}^2 dt,$$

$$\mathbf{SRP}_{\mathcal{L}}: (\hat{\boldsymbol{\alpha}}^{\hat{q}_2}, \hat{b}_0, \hat{b}_1) = \underset{\boldsymbol{\alpha}^{\hat{q}_2}, b_0, b_1}{\arg\min} \frac{1}{n_1} \sum_{i=1}^{n_1} \left\{ \tilde{X}_i(t_T) - \sum_{j=1}^{\hat{q}_2} \alpha_j^{\hat{q}_2} \tilde{X}_i(t_{T-j}) - \int_{t_0}^{t_{T-\hat{q}_2-1}} (b_0 + b_1 t) \tilde{X}_i(t) dt \right\}^2.$$

The objective function of our method ($\mathrm{SRP}_{\mathcal{C}}$) is in (3.9) and we determine $\hat{q}_1$ and $\hat{q}_2$ for MLR and $\mathrm{SRP}_{\mathcal{L}}$ by minimising $SIC(q)$ in (3.7) with appropriate $M(q)$ for each. In the implementation of FLR, we use cubic smoothing splines ($s = 3$) with the dimension $L = 35$ for both $\beta(t)$ and $X_i(t)$ where the derivative order of $\beta(t)$ is $m = 2$ and λ is selected by minimising GCV in (3.14). Ridge parameter is also optimised by minimising GCV. For the implementation of other methods, we follow the suggestions of each paper for selecting the tuning parameters and the R code is available on the web (FLiRTI: http://www-bcf.usc.edu/~gareth/research/Research.html, MPDP and NP: http://www.math.univ-toulouse.fr/~ferraty/). The R code for all simulations can be downloaded from our GitHub repository (Maeng, 2019b).

As an aside, we considered two variations of our proposal $\mathbf{SRP}_{\mathcal{C}}$ as follows:

$$(\hat{\boldsymbol{\alpha}}^q, \hat{\beta}^q(t)) = \underset{\boldsymbol{\alpha}^q, \beta^q(t)}{\arg\min} \left[\frac{1}{n} \sum_{i=1}^{n} \left\{ \tilde{X}_i(t_T) - \sum_{j=1}^{q} \alpha_j^q \tilde{X}_i(t_{T-j}) - \int_{t_0}^{t_{T-q-1}} \beta^q(t) \tilde{X}_i(t) dt \right\}^2 + \lambda_2 \int_{t_0}^{t_{T-q-1}} \left\{ \beta^{q(m)}(t) \right\}^2 dt \right], \tag{3.19}$$

$$(\hat{\boldsymbol{\alpha}}^q, \hat{\beta}^q(t)) = \underset{\boldsymbol{\alpha}^q, \beta^q(t)}{\arg\min} \left[\frac{1}{n} \sum_{i=1}^{n} \left\{ \tilde{X}_i(t_T) - \sum_{j=1}^{q} \alpha_j^q \tilde{X}_i(t_{T-j}) - \int_{t_0}^{t_{T-q-1}} \beta^q(t) \tilde{X}_i(t) dt \right\}^2 + \lambda_1 (\boldsymbol{\alpha}^q)^\top \boldsymbol{\alpha}^q + \lambda_2 \int_{t_0}^{t_{T-q-1}} \left\{ \beta^{q(m)}(t) \right\}^2 dt \right], \tag{3.20}$$

where the estimators in (3.19) are attained by penalising the smooth function $\beta(t)$ only, while those in (3.20) are affected also by a ridge-type penalty for scalar regression coefficients $\boldsymbol{\alpha}$. The only difference between our model $\mathbf{SRP}_{\mathcal{C}}$ and its variant in (3.20) is that $\mathbf{SRP}_{\mathcal{C}}$ deals with possible multicollinearity between both rough (scalar) and smooth (functional) coefficients, while the model in (3.20) only considers multicollinearity between rough coefficients. As our proposal showed better and more stable prediction performances than those two variants above, we only report the results of $\mathbf{SRP}_{\mathcal{C}}$.

3.4.2 Simulation results

The top row of Figure 3.6 shows that the mean of 100 $SIC(q)$ is minimised at true $q_0 = 3$ for all cases. Case 2 shows a more rapid decrease than the other cases when $q \uparrow q_0$ due to the larger size of change at the change-point. Similarly, in the bottom row, we see that the mode of $\hat{q}$ is $q_0 = 3$ in all cases. Since Cases 1 and 3 have a relatively smooth $\boldsymbol{\alpha}$, $\hat{q} = 1, 2 (< q_0)$ are selected more frequently than in Cases 2 and 4, which have relatively more fluctuating $\boldsymbol{\alpha}$'s. Figure 3.7 provides numerical evidence of the increased closeness of $\hat{q}$ to q_0 in Case 4 as the sample size n increases.

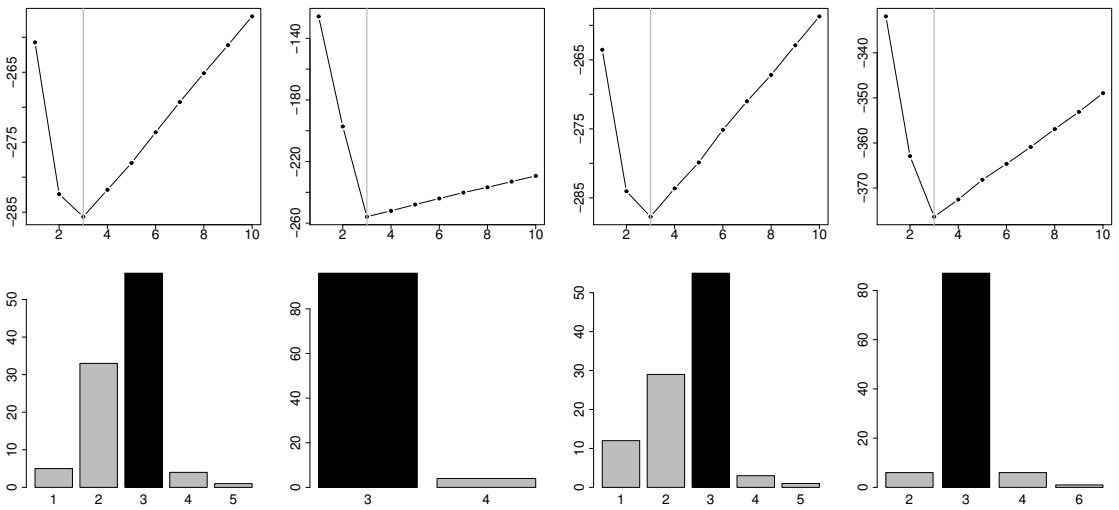


Fig. 3.6 (1^{st} row) Mean of $\{SIC(q)\}_{1\leq q\leq 10}$ defined in formula (3.7) over 100 simulation runs for Cases 1-4 (1^{st}-4^{th} column); (2^{nd} row) Barplots of the 100 $\hat{q}$ estimated by minimising $\{SIC(q)\}_{1\leq q\leq 30}$ where the black bars indicate the true change-point index parameter $q_0 = 3$.

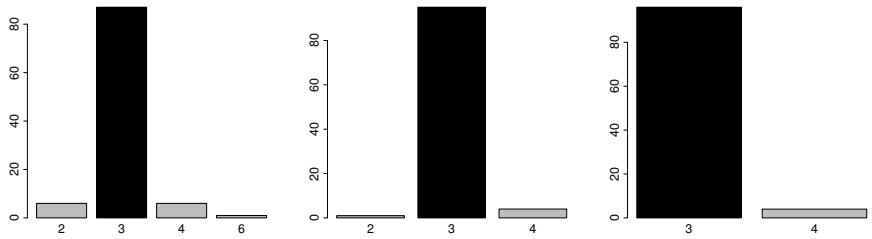


Fig. 3.7 Barplots of the 100 $\hat{q}$ estimated by minimising $SIC(q)$ in formula (3.7) with increasing $n = 300, 600, 1200$ under Case 4. The black bars indicate the true change-point index parameter $q_0 = 3$.

As is apparent from Table 3.1, FLR and RIDGE perform systematically worse than the others. Our proposal, $\text{SRP}_{\mathcal{C}}$, outperforms the others in Cases 1, 2 and 4 and the difference is the most striking in Cases 2 and 4, in which a sudden smoothness change occurs. In Case 3 whose true $\beta(t)$ is linear, $\text{SRP}_{\mathcal{L}}$ turns out to be the best-performing method.

Examining Figure 3.8, while the misestimation in $\text{SRP}_{\mathcal{C}}$ is mainly located around the true change-point, in FLiRTI and FLR it is scattered over the whole interval. In addition, the graph offers visual confirmation of the superior performance of $\text{SRP}_{\mathcal{C}}$ in

Table 3.1 The mean(sd) of SSE($\times 10^2$) defined in formula (3.17) over 100 simulation runs for the parametric methods in all cases. Bold: methods with the lowest mean of SSE.

Case	MLR	FLR	FLiRTI	$\text{SRP}_{\mathcal{L}}$	$\text{SRP}_{\mathcal{C}}$	RIDGE
1	1.39(0.73)	5.32(1.33)	1.11(0.44)	1.43(0.69)	**1.00**(0.86)	19.90(2.05)
2	10.24(2.59)	75.08(1.76)	31.25(9.24)	9.09(1.03)	**2.06**(0.76)	72.80(2.87)
3	0.79(0.55)	5.28(1.30)	0.78(0.39)	**0.64**(0.56)	1.08(1.20)	19.12(1.91)
4	11.31(1.38)	21.37(1.63)	9.72(2.32)	6.96(0.79)	**1.03**(0.44)	24.30(1.23)

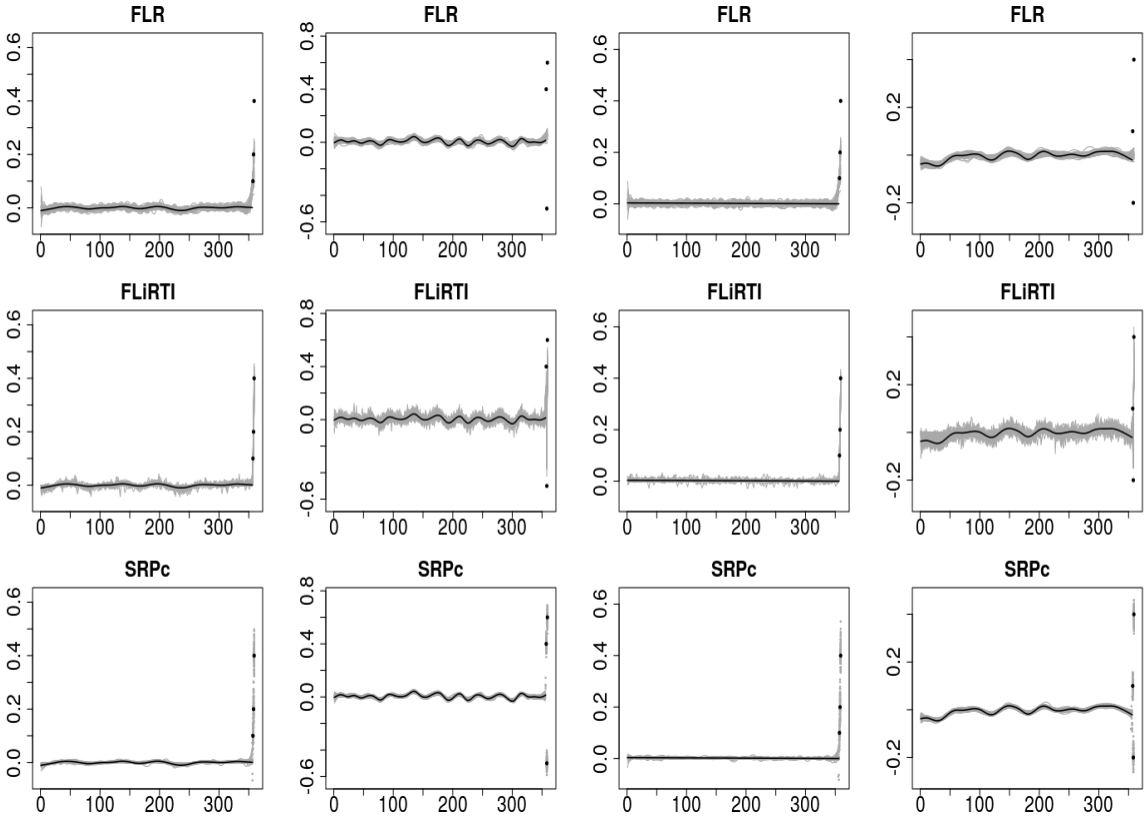


Fig. 3.8 True (black) and 100 estimated (grey) regression parameters for Cases 1-4($1^{st} - 4^{th}$ column) with three methods, FLR(1^{st} row), FLiRTI(2^{nd} row) and $\text{SRP}_{\mathcal{C}}$(3^{rd} row). The corresponding numerical summaries of these results are in Table 3.1.

Cases 1, 2 and 4. In particular, in Cases 2 and 4, FLR ignores the sudden fluctuation in $\boldsymbol{\alpha}$ by estimating it as a smooth function. Unlike FLR and FLiRTI, $\text{SRP}_{\mathcal{C}}$ shows its

advantages not only when scale changes are present (Cases 1 and 3) but also when a sudden smoothness change occurs at the change-point (Cases 2 and 4).

Table 3.2 contains two more columns than Table 3.1 as the mean-square prediction error can also be obtained for the nonparametric methods, MPDP and NP, which do not involve the estimation of $(\hat{\boldsymbol{\alpha}}, \hat{\beta}(t))$. In all cases considered, FLR, MPDP, NP and RIDGE show worse prediction performance than the other methods. $\text{SRP}_{\mathcal{C}}$ performs better than FLiRTI for all cases (but more noticeably so in Cases 2 and 4). $\text{SRP}_{\mathcal{C}}$ is superior to $\text{SRP}_{\mathcal{L}}$ in all cases except Case 3 which is expected since Case 3 includes a linear $\beta_0(t)$. However, $\text{SRP}_{\mathcal{C}}$ is not far behind $\text{SRP}_{\mathcal{L}}$ in Case 3 as the smoothness of $\hat{\beta}(t)$ is flexibly controlled by the automatically chosen penalty. We note that all cases considered in simulations assume a single change-point in the sequence of regression coefficients under the framework of decaying memory of time series. We expect that the $\text{SRP}_{\mathcal{C}}$ will underperform but the FLiRTI will outperform in the case when the sequence of true regression coefficients is a mix of zero regions and non-zero regions with more than one change-point.

Table 3.2 The mean(sd) of MSPE($\times 10^2$) defined in formula (3.18) over 100 simulation runs for all methods in all cases. Bold: methods with the lowest mean of MSPE.

Case	MLR	FLR	FLiRTI	$\text{SRP}_{\mathcal{L}}$	$\text{SRP}_{\mathcal{C}}$	MPDP	NP	RIDGE
1	21.83	23.39	20.48	22.12	**18.95**	26.22	79.04	43.52
	(2.7)	(3.2)	(2.7)	(2.8)	(3.2)	(5.1)	(9.9)	(7.0)
2	53.97	83.38	51.71	50.81	**27.55**	69.20	102.21	94.76
	(7.0)	(9.9)	(9.3)	(7.1)	(4.4)	(21.4)	(11.5)	(11.3)
3	17.26	22.01	17.86	**15.61**	16.80	21.41	74.82	41.35
	(2.1)	(3.0)	(2.5)	(2.1)	(3.3)	(3.8)	(9.7)	(6.8)
4	30.48	28.17	22.17	22.05	**10.88**	39.18	43.54	35.68
	(4.2)	(4.2)	(4.1)	(2.8)	(1.6)	(15.7)	(5.6)	(4.3)

From the viewpoint of choosing predictive design points (sometimes called points of impact by others), three elements in the true scalar coefficient vector, $\boldsymbol{\alpha}_0^\top$, can be

considered as a set of predictive points in all cases as they have a relatively larger size than $\beta_0(t)$. Among the candidate models, MPDP is established on this idea, thus we compare three predictive time-points (t_{359}, t_{358}, t_{357}) with those selected by MPDP although MPDP performs a nonparametric regression on the selected predictive design points.

Table 3.3 The percentages indicating how many time-points are selected as the most-predictive design points from (t_{359}, t_{358}, t_{357}) by MPDP over 100 simulation runs in all cases.

	Number of time-points chosen from (t_{359}, t_{358}, t_{357})			
Case	3 points selected	2 points selected	1 point selected	None
1	15%	84%	1%	0%
2	78%	3%	17%	2%
3	27%	73%	0%	0%
4	4%	48%	48%	0%

As shown in Table 3.3, all three points (t_{359}, t_{358}, t_{357}) are more often selected as the predictive design points when the size of change at the change-point is relatively large (e.g. in Case 2). Comparing the other three cases (Cases 1, 3 and 4), although they have the equal size of $\boldsymbol{\alpha}_0$ as $(|\alpha_{0,1}|, |\alpha_{0,2}|, |\alpha_{0,3}|) = (0.4, 0.2, 0.1)$, the percentage of choosing all three points is highest in Case 3 followed by Cases 1 and 4. This indicates that the smoothness of $\beta(t)$ is also an important factor for differentiating the non-influential part from the most informative points. In other words, under the equal size and the equal length of $\boldsymbol{\alpha}_0$, the more flat $\beta(t)$ is, the easier we detect the predictive points.

3.5 Data applications

In this section, our methodology is applied to country fertility data, Mexico city pollution data, stock volatility series and sunspot number data. The data can be obtained from the Human Fertility Database (www.humanfertility.org), the R package `aire.zmvm`, the Wharton Research Data Services (wrds-web.wharton.upenn.edu/wrds/) and the Base R datasets available from CRAN, respectively.

3.5.1 Country fertility rate data

Forecasting future fertility rates has a great impact on governments in planning children's service and education. We use fertility rates at age 20, recorded for 36 years from 1974 to 2009 for 31 countries around the world. As shown in Figure 3.9, the fertility rates at age 20 show an overall decreasing trend in all countries and although it is not illustrated in this section, similar patterns are observed at ages 21–26, while fertility rates at ages 30–39 have obvious increasing trends in recent years from 1990 onwards, which reflects the phenomenon of more women deferring childbirth to a later age.

The final observation recorded in 2009 is predicted from the past observations from 1974 to 2008. To compare the prediction power of the new model with competitors, we split the whole dataset into a training sample of size $n_1 = 26$ and a test set of size $n_2 = 5$ randomly 100 times and compute the mean, median and standard deviation of the 100 mean-square prediction errors defined in (3.18). In the training set, the B-spline expansion with dimension $L = 9$ is used for $\text{SRP}_{\mathcal{C}}$, $\text{SRP}_{\mathcal{L}}$ and FLR. As found in Table 3.4, MLR, $\text{SRP}_{\mathcal{C}}$ and $\text{SRP}_{\mathcal{L}}$ lead to similar performance in prediction, which is better than that of the nonparametric methods (MPDP, NP), the full functional model (FLR), the full scalar setting (RIDGE) and FLiRTI.

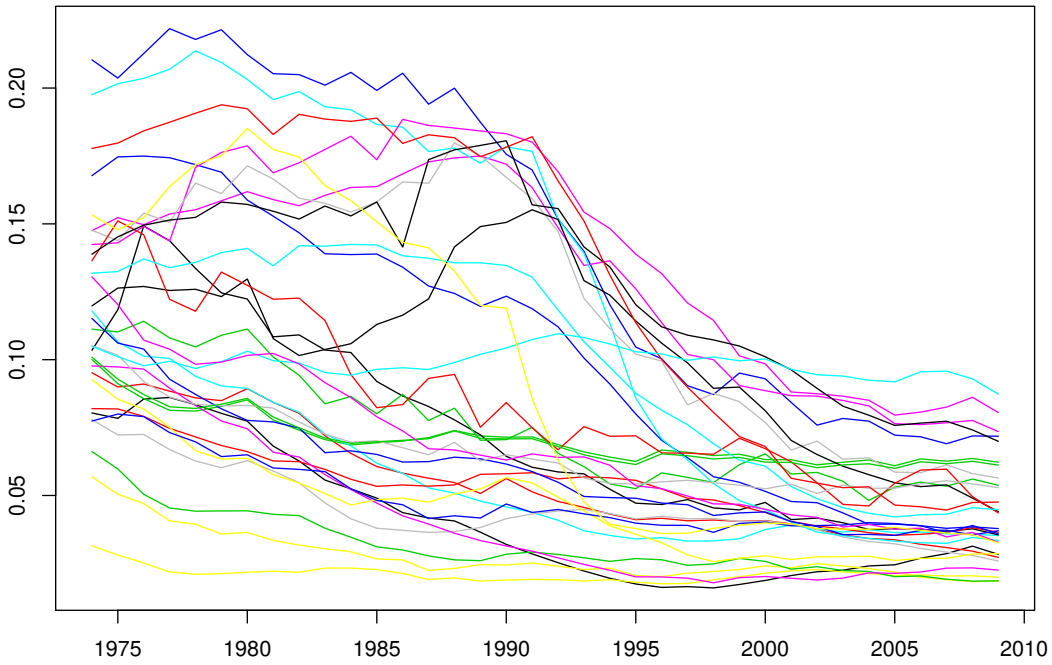


Fig. 3.9 The fertility rates at age 20 from 1974 to 2009 for 31 countries.

Table 3.4 The mean, median and standard deviation of 100 MSPE's ($\times 10^6$) defined in formula (3.18) for all methods described in Section 3.4.1, for the case study in Section 3.5.1. Bold: methods with the three lowest MSPE's.

	MLR	FLR	FLiRTI	$\text{SRP}_{\mathcal{L}}$	$\text{SRP}_{\mathcal{C}}$	MPDP	NP	RIDGE
mean	**3.36**	12.60	5.99	**3.45**	**3.73**	5.38	139.55	7.73
median	**2.98**	9.15	3.95	**3.12**	**3.28**	3.65	118.48	4.97
sd	2.00	10.70	6.13	1.94	2.32	5.33	114.58	7.39

As shown in Figure 3.10, $\hat{q}_\cdot = 1, 2$ are the most frequently selected as the optimal size of scalar variables for MLR, $\text{SRP}_{\mathcal{L}}$ and $\text{SRP}_{\mathcal{C}}$. Although MLR and $\text{SRP}_{\mathcal{L}}$ seem to be slightly better than $\text{SRP}_{\mathcal{C}}$ in prediction in Table 3.4, Figure 3.10 shows that $\text{SRP}_{\mathcal{C}}$ is the most frequently selected as the best-performing method in terms of MSPE from 100 samples. In Figure 3.11, the functional estimators $\hat{\beta}(t)$ for FLR and FLiRTI and the discrete ones for RIDGE live in the whole interval $t \in [t_0, t_{T-1}]$ while $\text{SRP}_{\mathcal{C}}$, MLR

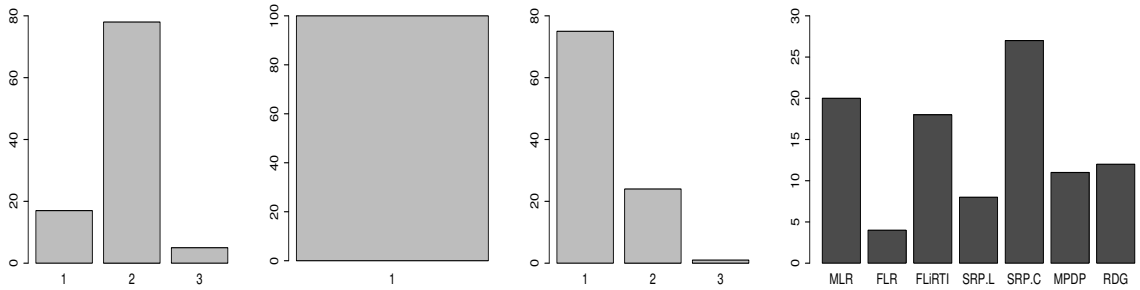


Fig. 3.10 Barplots of the 100 $\hat{q}_1$ for MLR (first), 100 $\hat{q}_2$ for $\text{SRP}_{\mathcal{L}}$ (second) and 100 $\hat{q}$ for $\text{SRP}_{\mathcal{C}}$ (third) estimated by minimising $\{SIC(q), 1 \leq q \leq 4\}$ in formula (3.7) and the frequency barplot of the best-performing method (with the lowest MSPE) out of the 100 samples (fourth) for the case study in Section 3.5.1.

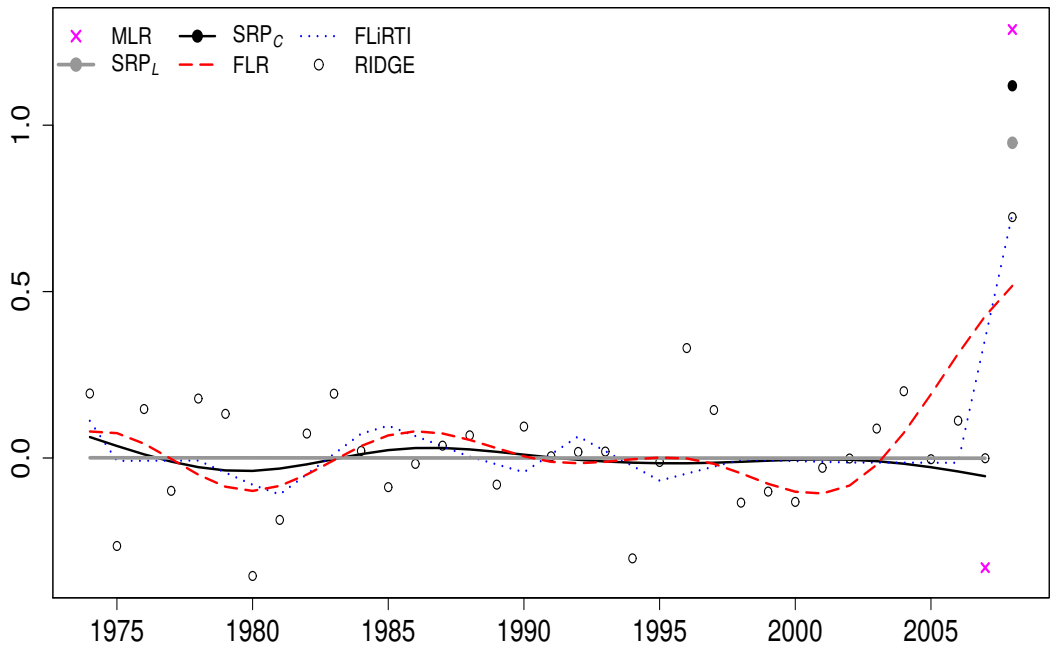


Fig. 3.11 A randomly selected estimated regression coefficients of the six parametric methods (MLR, $\text{SRP}_{\mathcal{L}}$, $\text{SRP}_{\mathcal{C}}$, FLR, FLiRTI, RIDGE) for predicting fertility rates at age 20 in 2009 from the past observations (1974-2008).

and $\text{SRP}_{\mathcal{L}}$ assign the corresponding subintervals for $\hat{\boldsymbol{\alpha}}$ with the optimally chosen $\hat{q} = 1$, $\hat{q}_1 = 2$ and $\hat{q}_2 = 1$ (respectively). The estimated curves for FLR and FLiRTI and the estimated coefficients for RIDGE appear to be relatively oscillatory over the entire

interval under a fixed smoothness while the smoothness of the SRP estimators varies as dictated by their design. Interestingly, all parametric methods give a large size of the regression coefficient at year 2008, which contrasts with the coefficients for years 1974–2007 which are close to zero. In a time series context, this indicates that the fertility rate in 2008 is more influential for predicting the fertility rate in 2009 than the older observations are.

3.5.2 Nitrogen oxides in Mexico City

We use the daily curves of hourly average nitrogen oxides level in Mexico City, introduced in Section 3.1. As shown in Figure 3.1, daily curves contain 24 observations each and have similar patterns including two peaks around hours 9 and 21. The final observation recorded at hour 24 is predicted from the past observations indexed 1 to 23. We split the whole dataset into a training sample of size $n_1 = 161$ and a test set of size $n_2 = 86$ randomly 100 times and compute the mean, median and standard deviation of the 100 mean-square prediction errors defined in (3.18). In the training set, the B-spline expansion with dimension $L = 9$ is used for $\text{SRP}_{\mathcal{C}}$, $\text{SRP}_{\mathcal{L}}$ and FLR. As found in Table 3.5 and Figure 3.12, $\text{SRP}_{\mathcal{C}}$ gives the best prediction among all methods and is also the most frequently selected as the best-performing one from the 100 samples in terms of MSPE. As shown in Figure 3.12, $\hat{q} = 3$ is the most frequently selected as the optimal size of scalar variables for $\text{SRP}_{\mathcal{C}}$ while $\hat{q}_{\cdot} = 2$ is so for MLR and $\text{SRP}_{\mathcal{L}}$.

In Figure 3.13, it is interesting to observe that the smooth portion of the SRP parameter vector appears to be non-trivially different from zero, which, together with the fact that the SRP model outperforms its competitors in the forecasting exercise reported above, provides evidence for the existence and impact of the long-term temporal dependence in this dataset. It is also apparent that all the methods attempt to fit a particularly large-size regression coefficient at hour 23. The $\text{SRP}_{\mathcal{C}}$ curve detects

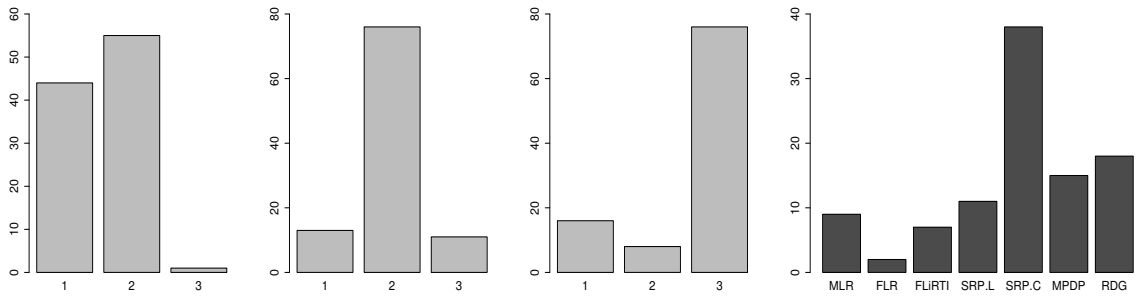


Fig. 3.12 Barplots of the 100 $\hat{q}_1$ for MLR (first), 100 $\hat{q}_2$ for SRP$_{\mathcal{L}}$ (second) and 100 $\hat{q}$ for SRP$_{\mathcal{C}}$ (third) estimated by minimising $\{SIC(q), 1 \leq q \leq 3\}$ in formula (3.7) and the frequency barplot of the best-performing method (with the lowest MSPE) out of the 100 samples (fourth) for the case study in Section 3.5.2.

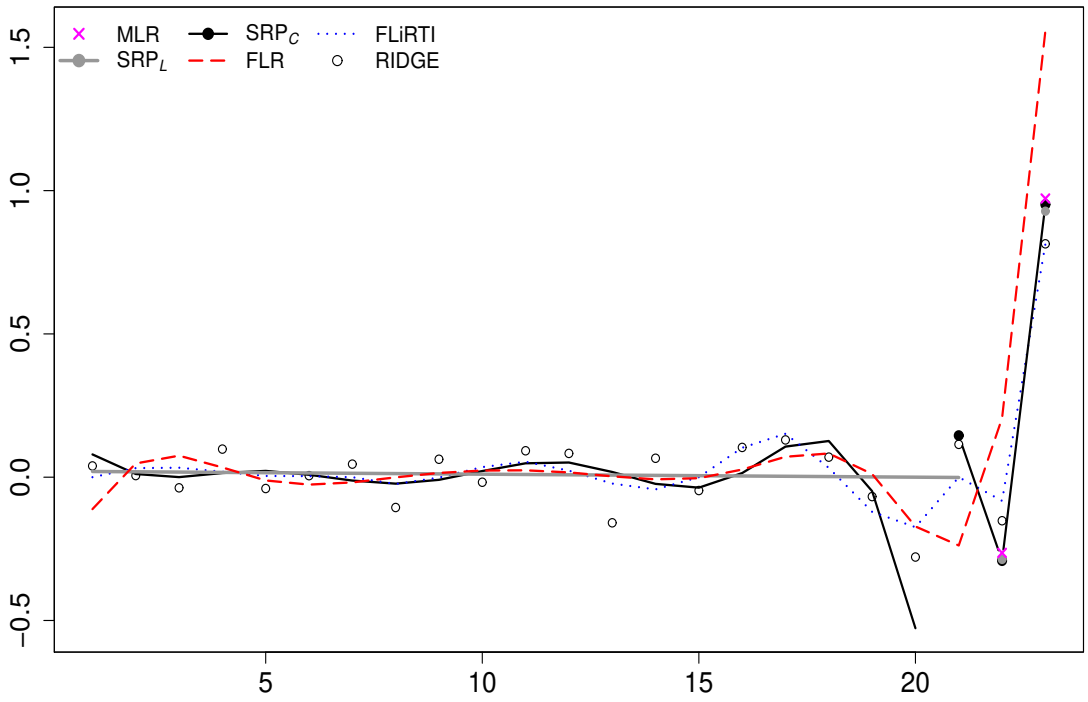


Fig. 3.13 A randomly selected estimated regression coefficients of the six parametric methods (MLR, SRP$_{\mathcal{L}}$, SRP$_{\mathcal{C}}$, FLR, FLiRTI, RIDGE) for predicting the average of nitrogen oxides level at hour 24.

a change at hour 20, where it experiences a seemingly non-trivial drop. It would be difficult for us to conclude that this drop is merely caused by a boundary effect as the RIDGE solution (in which there are no boundary effects to speak of) also experiences

a dip at that point. In the same manner, the sudden increase observed in the FLR curve at hour 23 does not appear to be a mere boundary effect, but it also reflects this method's own effort to fit the influential predictor under its own smoothness constraints. The results in Table 3.5 show that it is useful to apply two different regularisations, as done in $\text{SRP}_{\mathcal{C}}$, depending on the perceived importance of predictors, rather than estimating the regression coefficients under an unvarying regularisation, as done in RIDGE.

Table 3.5 The mean, median and standard deviation of 100 MSPE's ($\times 10^2$) defined in formula (3.18) for all methods described in Section 3.4.1, for the case study in Section 3.5.2. Bold: methods with the three lowest MSPE's.

	MLR	FLR	FLiRTI	$\text{SRP}_{\mathcal{L}}$	$\text{SRP}_{\mathcal{C}}$	MPDP	NP	RIDGE
mean	75.50	86.44	**73.88**	75.41	**72.35**	74.92	126.09	**74.42**
median	75.38	85.16	**74.04**	75.13	**71.84**	74.23	126.99	**73.41**
sd	12.92	14.03	12.96	14.10	13.18	13.13	26.63	12.94

3.5.3 High frequency volatility series

In financial data analysis, modelling high-frequency volatility has attracted much attention in recent years. Especially, in the functional framework, nonparametric methods have been extensively studied (Bandi and Phillips, 2003; Kristensen, 2010; Reno, 2008). Müller et al. (2011) emphasise the random nature of volatility functions under the assumption that the repeated realisations of the volatility trajectories come from a suitable functional volatility process. Our interest is also in the random nature of functional observations rather than in modelling potential dependencies between curves, therefore, as in Müller et al. (2011), we view the daily curves as i.i.d. random functions. We aim to predict the latest point of the curves from the past observations.

Specifically, our methodology is applied to the prediction of the Disney stock volatility where the raw observations contain $n = 248$ trading days available from

January 2, 2013 to December 30, 2013 and each curve has 395 grid points of closing prices recorded every 1 minute. The volatility trajectories are obtained from the return series in the same way as in Müller et al. (2011), however we retain the roughness of volatility trajectories by using natural cubic splines as in (3.5) rather than smoothing them. This is important as volatility is not observable but typically estimated to be oscillatory, thus an extra smoothing step can possibly cause the loss of important information as stated in Kneip et al. (2016).

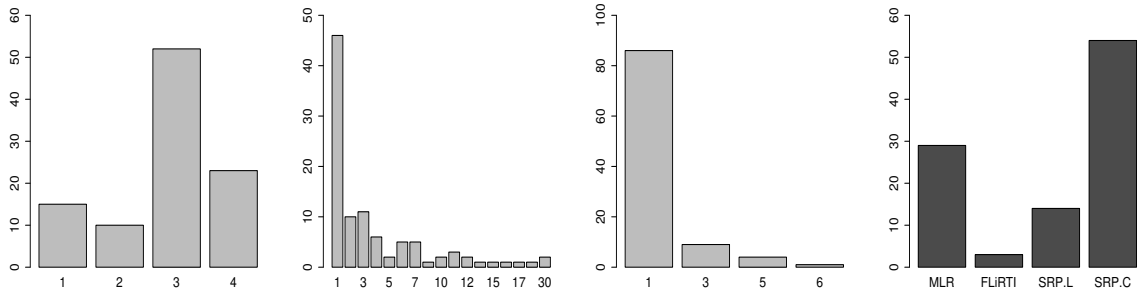


Fig. 3.14 Barplots of the 100 $\hat{q}_1$ for MLR (first), 100 $\hat{q}_2$ for $\text{SRP}_{\mathcal{L}}$ (second) and 100 $\hat{q}$ for $\text{SRP}_{\mathcal{C}}$ (third) estimated by minimising $\{SIC(q), 1 \leq q \leq 30\}$ in formula (3.7) and the frequency barplot of the best-performing method (with the lowest MSPE) out of the 100 samples (fourth) for the case study in Section 3.5.3.

We split the dataset into a training and a test set of size $n_1 = n_2 = 124$ randomly 100 times and in the training set, the B-spline expansion with dimension $L = 35$ is used for $\text{SRP}_{\mathcal{C}}$, $\text{SRP}_{\mathcal{L}}$ and FLR. Figure 3.14 shows that $\hat{q}_1 = 3$ is the most frequently chosen for MLR while $\hat{q}_2 = 1$ and $\hat{q} = 1$ are the most frequently selected for $\text{SRP}_{\mathcal{L}}$ and $\text{SRP}_{\mathcal{C}}$, respectively.

Similar to the previous examples in Sections 3.5.1 and 3.5.2, Figure 3.15 shows that all the parametric methods reflect the 'fading memory' of the time series by assigning a large-size regression coefficient for observations located close to the closing volatilities, which contrasts with the coefficients for intervals positioned far from the closing volatility. As found in Table 3.6 and Figure 3.14, $\text{SRP}_{\mathcal{C}}$ leads to the best prediction

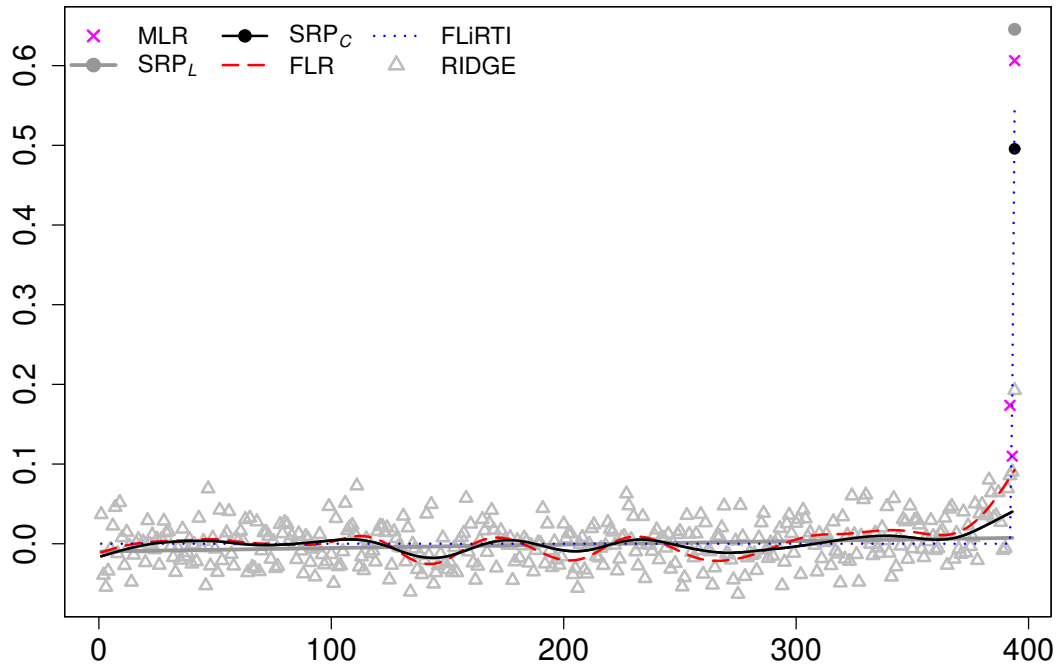


Fig. 3.15 A randomly selected estimated regression coefficients of the six parametric methods (MLR, SRP$_{\mathcal{L}}$, SRP$_{\mathcal{C}}$, FLR, FLiRTI, RIDGE) for predicting closing volatility of the Disney stock data from January to December in 2013.

among all methods and is also the most frequently selected as the best-performing one in terms of MSPE from 100 samples.

Table 3.6 The mean, median and standard deviation of 100 MSPE's defined in formula (3.18) for all methods described in Section 3.4.1, for the case study from Section 3.5.3. Bold: methods with the three lowest MSPE's.

	MLR	FLR	FLiRTI	SRP$_{\mathcal{L}}$	SRP$_{\mathcal{C}}$	MPDP	NP	RIDGE
mean	**2.88**	4.10	3.13	**2.96**	**2.78**	3.02	6.29	4.34
median	**2.80**	4.05	3.08	2.91	**2.72**	**2.77**	6.18	4.29
sd	0.56	0.58	0.68	0.56	0.51	1.52	0.71	0.48

3.5.4 Monthly numbers of sunspots

In this section, we demonstrate the usefulness the SRP framework in univariate time series modelling, as an alternative to the AR model, which is often used in time series forecasting. The SRP model is similar to the AR model in that they both specify the fading memory structure of the time series under linear dependence of the output variable on its own previous values. In practice, the AR(p) model is usually fitted with a small p for simplicity, interpretability and better forecasting performance, however it may fail in the presence of longer memory. In this case, the SRP model can also be used for the forecasting of a univariate time series, where it becomes an autoregressive (AR) model with a large order (e.g. AR(T) in (3.3) with a fixed T) under the smooth-rough regularisation.

We use the sunspot number data shown in Figure 3.2 in Section 3.1. The data contains 3177 observations available from 1749 to 2013 and we perform a square root transformation to the raw data. We split the whole dataset into a training sample of size $n_1 = 2223$ and a test set of size $n_2 = 954$ and create the data matrix for each set via a moving window with a prespecified number $T+1 = 151$ of discrete points in one curve (150 for covariates and 1 for the response variable), i.e. from the univariate time series $(x_1, x_2, \ldots, x_{n_1})$ in the training sample, we create 2073 curves, $X_1(t) = (x_1, x_2, \ldots, x_{151})$, $X_2(t) = (x_2, x_3, \ldots, x_{152})$, …, $X_{n_1-151+1} = (x_{n_1-150}, x_{n_1-149}, \ldots, x_{n_1})$. In the same way, we create 804 curves for the test sample. In each curve, we use the last points as the response variable and the covariates are the remaining 150 observations. Due to the temporal dependence in the entire dataset, we do not randomly repeat the construction of the training and test sets.

From the training set, with $L = 35$, the optimal change-point index parameter for MLR, $\text{SRP}_{\mathcal{L}}$ and $\text{SRP}_{\mathcal{C}}$ are chosen as $\hat{q}_1 = 5, \hat{q}_2 = 6, \hat{q} = 2$ (respectively) from $\{q : 1 \leq q \leq 15\}$ as shown in Figure 3.16. As the optimal size $\hat{q}_1 = 5$ for MLR is

obtained by minimising the SIC, the estimated regression coefficients are very close to that of the AR(5) model and the significance of the first five lags is already revealed in the partial autocorrelation function in Figure 3.2. In Figure 3.16, the FLR and RIDGE estimators appear to be relatively oscillatory over the entire interval, while the estimators for FLiRTI and $\text{SRP}_{\mathcal{C}}$ are relatively smoother. We also obtain the OLS (ordinary least-squares) estimator which is slightly more fluctuating than RIDGE, but is not included in Figure 3.16. As is apparent from Table 3.7, our approach shows an improvement in prediction compared to the other methods. From this example, $\text{SRP}_{\mathcal{C}}$ appears to be a useful substitution for a classical AR(p) model with a small p, especially when the memory of a time series is relatively long.

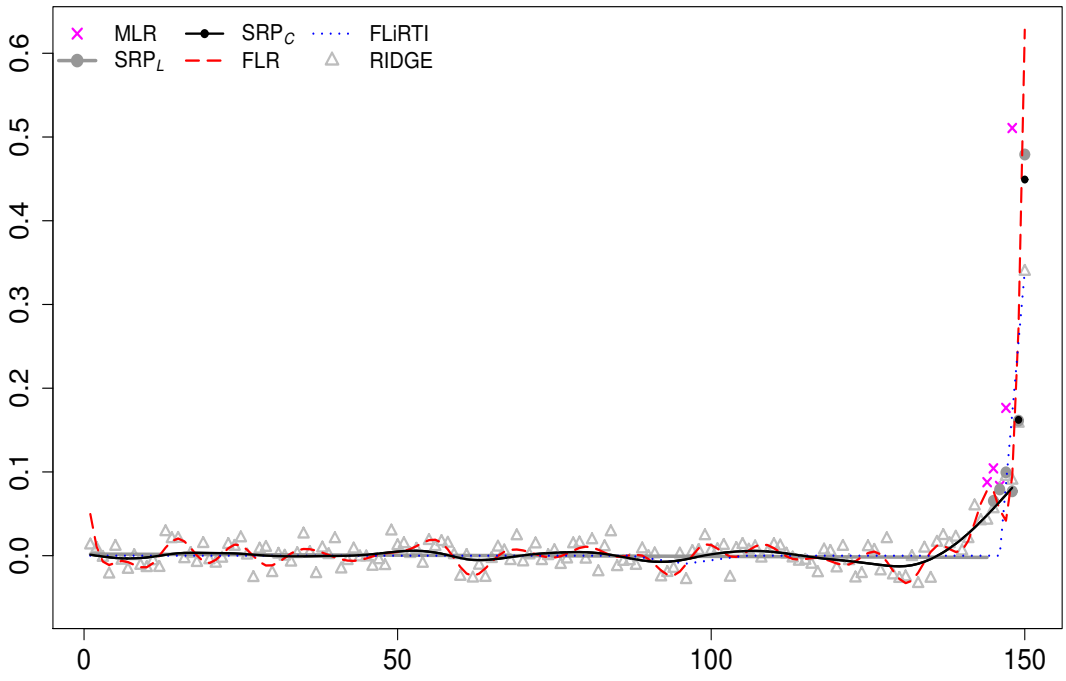


Fig. 3.16 Estimated regression coefficients of the six parametric methods (MLR, $\text{SRP}_{\mathcal{L}}, \text{SRP}_{\mathcal{C}}$, FLR, FLiRTI, RIDGE) for predicting the sunspot number of next month from past 150 months of sunspot number.

Table 3.7 MSPE ($\times 10^2$) defined in formula (3.18) for all parametric methods described in Section 3.4.1 and OLS, for the case study from Section 3.5.4. Bold: methods with the three lowest MSPE's.

	MLR	FLR	FLiRTI	$\text{SRP}_{\mathcal{L}}$	$\text{SRP}_{\mathcal{C}}$	RIDGE	OLS
MSPE	11.67	12.09	12.59	**11.09**	**10.72**	11.17	**11.11**

3.6 Proofs

The proof of Theorem 3.1 in Section 3.3 is presented. The preparatory lemma is developed first and the main part of the proof is presented in Section 3.6.1.

Lemma 3.1 *Let $1 \leq q_1 < q_0$ as in Assumption 3.2. If Assumptions 3.1, 3.2, 3.5 and 3.6 hold then, uniformly in $q \in [q_1, q_0)$,*

$$(a) \quad \frac{1}{n}\sum_{i=1}^{n}(\varepsilon_i - \bar{\varepsilon})\left\{\sum_{j=1}^{q}(\alpha_{0,j} - \hat{\alpha}_j^q)\tilde{X}_i(t_{T-j})\right\} = O_p(n^{-1/2}|q|),$$

$$(b) \quad \frac{1}{n}\sum_{i=1}^{n}(\varepsilon_i - \bar{\varepsilon})\left\{\sum_{j=q_0+1}^{T}(\beta_0(t_{T-j}) - \hat{\beta}^q(t_{T-j}))\tilde{X}_i(t_{T-j})\right\} = O_p(n^{-1/2}|T - q_0|),$$

$$(c) \quad \frac{1}{n}\sum_{i=1}^{n}(\varepsilon_i - \bar{\varepsilon})\left\{\sum_{j=q+1}^{q_0}(\alpha_{0,j} - \hat{\beta}^q(t_{T-j}))\tilde{X}_i(t_{T-j})\right\} = O_p(n^{-1/2}|q_0 - q|).$$

Our Lemma 3.1 is similar to the Lemma in Hall and Hooker (2016) who study the consistency of truncation point in functional linear regression with one functional predictor. The proof of Lemma 3.1 can be simply obtained by following the proof of Lemma presented in the technical appendix to Hall and Hooker (2016) and also by considering a discrete version of it, i.e. replacing a curve with a vector, under our assumptions.

3.6.1 Proof of Theorem 3.1

Let q_1 and q_2 as in Assumptions 3.2 and 3.3, respectively. Since $X_i(t_T) = \mu + \sum_{j=1}^{q_0} \alpha_{0,j}\{X_i(t_{T-j}) - EX(t_{T-j})\} + \sum_{j=q_0+1}^{T} \beta_0(t_{T-j})\{X_i(t_{T-j}) - EX(t_{T-j})\} + \varepsilon_i$, we have $X_i(t_T) - \bar{X}(t_T) = \sum_{j=1}^{q_0} \alpha_{0,j}\tilde{X}_i(t_{T-j}) + \sum_{j=q_0+1}^{T} \beta_0(t_{T-j})\tilde{X}_i(t_{T-j}) + (\varepsilon_i - \bar{\varepsilon})$, thus $M(q)$ defined in (3.8) of Section 3.2.1 is expanded as

$$\begin{aligned}
M(q) &= \frac{1}{n}\sum_{i=1}^{n}\left[X_i(t_T) - \hat{\mu}^q - \sum_{j=1}^{q}\hat{\alpha}_j^q X_i(t_{T-j}) - \sum_{j=q+1}^{T}\hat{\beta}^q(t_{T-j})X_i(t_{T-j})\right]^2 \\
&= \frac{1}{n}\sum_{i=1}^{n}\left[X_i(t_T) - \bar{X}(t_T) - \sum_{j=1}^{q}\hat{\alpha}_j^q \tilde{X}_i(t_{T-j}) - \sum_{j=q+1}^{T}\hat{\beta}^q(t_{T-j})\tilde{X}_i(t_{T-j})\right]^2 \\
&= \frac{1}{n}\sum_{i=1}^{n}\left[\sum_{j=1}^{q_0}\alpha_{0,j}\tilde{X}_i(t_{T-j}) + \sum_{j=q_0+1}^{T}\beta_0(t_{T-j})\tilde{X}_i(t_{T-j}) - \sum_{j=1}^{q}\hat{\alpha}_j^q\tilde{X}_i(t_{T-j})\right. \\
&\qquad \left. - \sum_{j=q+1}^{T}\hat{\beta}^q(t_{T-j})\tilde{X}_i(t_{T-j}) + (\varepsilon_i - \bar{\varepsilon})\right]^2,
\end{aligned}$$

where $q \in [q_1, q_2]$. $M(q)$ has a different form for three cases: 1) $q > q_0$, 2) $q < q_0$ and 3) $q = q_0$. Firstly, if $q > q_0$, for $q \in (q_0, q_2]$, we have

$$\begin{aligned}
M(q) = \frac{1}{n}\sum_{i=1}^{n}\left[\sum_{j=1}^{q_0}(\alpha_{0,j} - \hat{\alpha}_j^q)\tilde{X}_i(t_{T-j}) + \sum_{j=q+1}^{T}(\beta_0(t_{T-j}) - \hat{\beta}^q(t_{T-j}))\tilde{X}_i(t_{T-j})\right. \\
\left. + \sum_{j=q_0+1}^{q}(\beta_0(t_{T-j}) - \hat{\alpha}_j^q)\tilde{X}_i(t_{T-j}) + (\varepsilon_i - \bar{\varepsilon})\right]^2. \qquad (3.21)
\end{aligned}$$

If $q < q_0$, for $q \in [q_1, q_0)$,

$$\begin{aligned}
M(q) = \frac{1}{n}\sum_{i=1}^{n}\left[\sum_{j=1}^{q}(\alpha_{0,j} - \hat{\alpha}_j^q)\tilde{X}_i(t_{T-j}) + \sum_{j=q_0+1}^{T}(\beta_0(t_{T-j}) - \hat{\beta}^q(t_{T-j}))\tilde{X}_i(t_{T-j})\right. \\
\left. + \sum_{j=q+1}^{q_0}(\alpha_{0,j} - \hat{\beta}^q(t_{T-j}))\tilde{X}_i(t_{T-j}) + (\varepsilon_i - \bar{\varepsilon})\right]^2. \qquad (3.22)
\end{aligned}$$

Lastly, when $q = q_0$,

$$M(q) = \frac{1}{n}\sum_{i=1}^{n}\left[\sum_{j=1}^{q_0}(\alpha_{0,j} - \hat{\alpha}_j^{q_0})\tilde{X}_i(t_{T-j}) + \sum_{j=q_0+1}^{T}(\beta_0(t_{T-j}) - \hat{\beta}^{q_0}(t_{T-j}))\tilde{X}_i(t_{T-j}) + (\varepsilon_i - \bar{\varepsilon})\right]^2. \tag{3.23}$$

3.6.1.1 Convergence rates of $M(q)$ for three cases

Now we explore the behaviour of $M(q)$. For the first case, 1) $q > q_0$, under Assumptions 3.3 and 3.4, (3.21) simplifies to

$$\begin{aligned}
M(q) =& \frac{1}{n}\sum_{i=1}^{n}\left\{\sum_{j=1}^{q_0}(\alpha_{0,j} - \hat{\alpha}_j^{q})\tilde{X}_i(t_{T-j})\right\}^2 + \frac{1}{n}\sum_{i=1}^{n}\left\{\sum_{j=q+1}^{T}(\beta_0(t_{T-j}) - \hat{\beta}^{q}(t_{T-j}))\tilde{X}_i(t_{T-j})\right\}^2 \\
&+ \frac{1}{n}\sum_{i=1}^{n}\left\{\sum_{j=q_0+1}^{q}(\beta_0(t_{T-j}) - \hat{\alpha}_j^{q})\tilde{X}_i(t_{T-j})\right\}^2 \\
&+ \frac{2}{n}\sum_{i=1}^{n}(\varepsilon_i - \bar{\varepsilon})\left\{\sum_{j=1}^{q_0}(\alpha_{0,j} - \hat{\alpha}_j^{q})\tilde{X}_i(t_{T-j})\right\} \\
&+ \frac{2}{n}\sum_{i=1}^{n}(\varepsilon_i - \bar{\varepsilon})\left\{\sum_{j=q+1}^{T}(\beta_0(t_{T-j}) - \hat{\beta}^{q}(t_{T-j}))\tilde{X}_i(t_{T-j})\right\} \\
&+ \frac{2}{n}\sum_{i=1}^{n}(\varepsilon_i - \bar{\varepsilon})\left\{\sum_{j=q_0+1}^{q}(\beta_0(t_{T-j}) - \hat{\alpha}_j^{q})\tilde{X}_i(t_{T-j})\right\} + \frac{1}{n}\sum_{i=1}^{n}(\varepsilon_i - \bar{\varepsilon})^2 \\
=& O_p(1/n) + \mathbf{V}_n,
\end{aligned} \tag{3.24}$$

uniformly in $q \in (q_0, q_2]$, where $\mathbf{V}_n$ refers to the error term which does not depend on q. In the second case, 2) $q < q_0$, using Lemma 3.1, (3.22) simplifies to

$$\begin{aligned}
M(q) =& \frac{1}{n}\sum_{i=1}^{n}\left\{\sum_{j=1}^{q}(\alpha_{0,j}-\hat{\alpha}_j^q)\tilde{X}_i(t_{T-j})\right\}^2 + \frac{1}{n}\sum_{i=1}^{n}\left\{\sum_{j=q_0+1}^{T}(\beta_0(t_{T-j})-\hat{\beta}^q(t_{T-j}))\tilde{X}_i(t_{T-j})\right\}^2 \\
&+ \frac{1}{n}\sum_{i=1}^{n}\left\{\sum_{j=q+1}^{q_0}(\alpha_{0,j}-\hat{\beta}^q(t_{T-j}))\tilde{X}_i(t_{T-j})\right\}^2 \\
&+ \frac{2}{n}\sum_{i=1}^{n}(\varepsilon_i-\bar{\varepsilon})\left\{\sum_{j=1}^{q}(\alpha_{0,j}-\hat{\alpha}_j^q)\tilde{X}_i(t_{T-j})\right\} \\
&+ \frac{2}{n}\sum_{i=1}^{n}(\varepsilon_i-\bar{\varepsilon})\left\{\sum_{j=q_0+1}^{T}(\beta_0(t_{T-j})-\hat{\beta}^q(t_{T-j}))\tilde{X}_i(t_{T-j})\right\} \\
&+ \frac{2}{n}\sum_{i=1}^{n}(\varepsilon_i-\bar{\varepsilon})\left\{\sum_{j=q+1}^{q_0}(\alpha_{0,j}-\hat{\beta}^q(t_{T-j}))\tilde{X}_i(t_{T-j})\right\} + \frac{1}{n}\sum_{i=1}^{n}(\varepsilon_i-\bar{\varepsilon})^2 \\
=& M_1(q)+M_2(q)+M_3(q)+O_p(n^{-1/2}|q|)+O_p(n^{-1/2}|T-q_0|) \\
&+ O_p(n^{-1/2}|q_0-q|)+\mathbf{V}_n, \qquad (3.25)
\end{aligned}$$

uniformly in $q \in [q_1, q_0)$, where

$$M_1(q) = \sum_{1\le k_1,k_2\le q}\{\alpha_{0,k_1}-\hat{\alpha}_{k_1}^q\}\{\alpha_{0,k_2}-\hat{\alpha}_{k_2}^q\}\hat{K}_{(k_1,k_2)}, \qquad (3.26)$$

$$M_2(q) = \sum_{q_0+1\le k_1,k_2\le T}\{\beta_0(t_{T-k_1})-\hat{\beta}^q(t_{T-k_1})\}\{\beta_0(t_{T-k_2})-\hat{\beta}^q(t_{T-k_2})\}\hat{K}_{(k_1,k_2)}, \qquad (3.27)$$

$$M_3(q) = \sum_{q+1\le k_1,k_2\le q_0}\{\alpha_{0,k_1}-\hat{\beta}^q(t_{T-k_1})\}\{\alpha_{0,k_2}-\hat{\beta}^q(t_{T-k_2})\}\hat{K}_{(k_1,k_2)}, \qquad (3.28)$$

and $\hat{K}_{(k_1,k_2)}$ is the empirical version of K defined in Assumption 3.6. Now we define

$$\kappa_3(q) = \sum_{q+1\le k_1,k_2\le q_0}\{\alpha_{0,k_1}-\hat{\beta}^q(t_{T-k_1})\}\{\alpha_{0,k_2}-\hat{\beta}^q(t_{T-k_2})\}K_{(k_1,k_2)},$$

to deal with $M_3(q)$. If we show that, for any bounded vector $\boldsymbol{z} = (z_0, ..., z_{T-1})^\top$,

$$\sup_{u,v\in[0,T-1]} \left| \sum_{k_1=u}^{v} \sum_{k_2=u}^{v} z_{k_1} z_{k_2} \left\{ \hat{K}_{(k_1,k_2)} - K_{(k_1,k_2)} \right\} \right| \to 0 \quad \text{in probability,} \tag{3.29}$$

then we can argue that $\sup_{q\in[q_1,q_0)} \left| M_3(q) - \kappa_3(q) \right| \to 0$ in probability by taking a vector $\boldsymbol{z}$ with its elements $z_j = (\alpha_{0,j} - \hat{\beta}^q(t_{T-j}))$ if $q+1 \le j \le q_0$ and $z_j = 0$ otherwise. We can simply derive (3.29) under Assumptions 3.2 and 3.5 and the following inequality used in Hall and Hooker (2016),

$$\sup_{u,v\in[0,T-1]} \left| \sum_{k_1=u}^{v} \sum_{k_2=u}^{v} z_{k_1} z_{k_2} \left\{ \hat{K}_{(k_1,k_2)} - K_{(k_1,k_2)} \right\} \right| \le (\sup_j |z_j|)^2 \sum_{k_1=u}^{v} \sum_{k_2=u}^{v} \left| \hat{K}_{(k_1,k_2)} - K_{(k_1,k_2)} \right| \tag{3.30}$$

where $u, v \in [0, T-1]$. Similarly, $\kappa_1(q)$ and $\kappa_2(q)$ can be defined for $M_1(q)$ and $M_2(q)$, respectively and following from Assumption 3.2, $\kappa_1(q)$, $\kappa_2(q)$ and $\kappa_3(q)$ are bounded away from zero whenever $q < q_0$.

Lastly, when $q = q_0$, under Assumptions 3.3 and 3.4, (3.23) can be simplified as

$$\begin{aligned} M(q) =& \frac{1}{n} \sum_{i=1}^{n} \left\{ \sum_{j=1}^{q_0} (\alpha_{0,j} - \hat{\alpha}_j^{q_0}) \tilde{X}_i(t_{T-j}) \right\}^2 \\ &+ \frac{1}{n} \sum_{i=1}^{n} \left\{ \sum_{j=q_0+1}^{T} (\beta_0(t_{T-j}) - \hat{\beta}^{q_0}(t_{T-j})) \tilde{X}_i(t_{T-j}) \right\}^2 \\ &+ \frac{2}{n} \sum_{i=1}^{n} (\varepsilon_i - \bar{\varepsilon}) \left\{ \sum_{j=1}^{q_0} (\alpha_{0,j} - \hat{\alpha}_j^{q_0}) \tilde{X}_i(t_{T-j}) \right\} \\ &+ \frac{2}{n} \sum_{i=1}^{n} (\varepsilon_i - \bar{\varepsilon}) \left\{ \sum_{j=q_0+1}^{T} (\beta_0(t_{T-j}) - \hat{\beta}^{q_0}(t_{T-j})) \tilde{X}_i(t_{T-j}) \right\} + \frac{1}{n} \sum_{i=1}^{n} (\varepsilon_i - \bar{\varepsilon})^2 \\ =& O_p(1/n) + \mathbf{V}_n. \end{aligned} \tag{3.31}$$

3.6.1.2 Expansions of $SIC(q)$ based on $M(q)$

To prove Theorem 3.1, it suffices to show that $SIC(q) - SIC(q_0)$ is positive for both cases 1) $q > q_0$ and 2) $q < q_0$. If $q > q_0$, for $\epsilon > 0$,

$$\begin{aligned} SIC(q) - SIC(q_0) =& n \cdot \log\left(\frac{M(q)}{M(q_0)}\right) + (q - q_0) \cdot \log n \\ =& n \cdot \log\left(1 - \frac{M(q_0) - M(q)}{M(q_0)}\right) + (q - q_0) \cdot \log n \\ \geq& -n(1+\epsilon)\left(\frac{M(q_0) - M(q)}{M(q_0)}\right) + (q - q_0) \cdot \log n. \end{aligned}$$

where the last inequality is obtained from $\log(1-x) = -x - x^2/2 - x^3/3 - x^4/4 \cdots$. Since $\mathbf{V}_n = \sigma^2 + o_p(1)$ and $M(q_0) - M(q) = O_p(1/n)$ for $q > q_0$ by (3.24) and (3.31), $SIC(q) - SIC(q_0)$ is guaranteed to be positive as $n \to \infty$.

Conversely, if $q < q_0$,

$$\begin{aligned} SIC(q) - SIC(q_0) =& n \cdot \log\left(\frac{M(q)}{M(q_0)}\right) + (q - q_0) \cdot \log n \\ \geq& n \cdot \log\left(\frac{M(q)}{M(q_0)}\right) - q_0 \cdot \log n. \end{aligned}$$

Since it can be simply shown that $\frac{M(q)}{M(q_0)} > 1 + \frac{1}{n}$ for $q < q_0$ from (3.25) and (3.31), $SIC(q) - SIC(q_0)$ is guaranteed to be positive as $n \to \infty$. Hence, we simply deduce that $P(\hat{q} = q_0) \to 1$ as $n \to \infty$.

Chapter 4

Trend Segmentation in data sequences

4.1 Introduction

This chapter considers the change-point model

$$X_t = f_t + \varepsilon_t, \quad t = 1, \ldots, T, \tag{4.1}$$

where f_t is a deterministic and piecewise-linear signal containing N change-points, i.e. time indices at which the slope and/or the intercept in f_t undergoes changes. These changes occur at unknown locations $\eta_1, \eta_2, \ldots, \eta_N$. The ε_t's are iid random errors following the normal distribution with mean zero and variance σ^2. Both continuous and discontinuous changes in the linear trend are permitted. A point anomaly can be viewed as a separate data segment containing only one data point. Therefore, if f_{η_ℓ} is a point anomaly, then the two consecutive change-points that define it, $\eta_{\ell-1}$ and η_ℓ, are linked via $\eta_{\ell-1} = \eta_\ell - 1$ under the definition of a change-point specified later in (4.35). Our main interest is in the estimation of N and $\eta_1, \eta_2, \ldots, \eta_N$ under some assumptions

that quantify the difficulty of detecting each η_i; therefore, our aim is to segment the data into sections of linearity and/or point anomalies in f_t. In particular, a point anomaly can only be detected when it has a large enough jump size with respect to the signal levels to its right and left, while a change-point capturing a small size of linear trend change requires a longer distance from its adjacent change-points to be detected. Detecting both linear trend changes and point anomalies is an important applied problem in a variety of fields, for example Figure 4.1 shows a land temperature dataset; some strong local trends appear to be present and the point corresponding to 1918 appears to be a point anomaly. Regarding the 1918 observation, Moore and Babij (2017) report that "[t]he winter of 1917/1918 is referred to as the Great Frost Winter in Iceland. It was the coldest winter in the region during the twentieth century. It was remarkable for the presence of sea ice in Reykjavik Harbour as well as for the unusually large number of polar bear sightings in northern Iceland." As illustrated

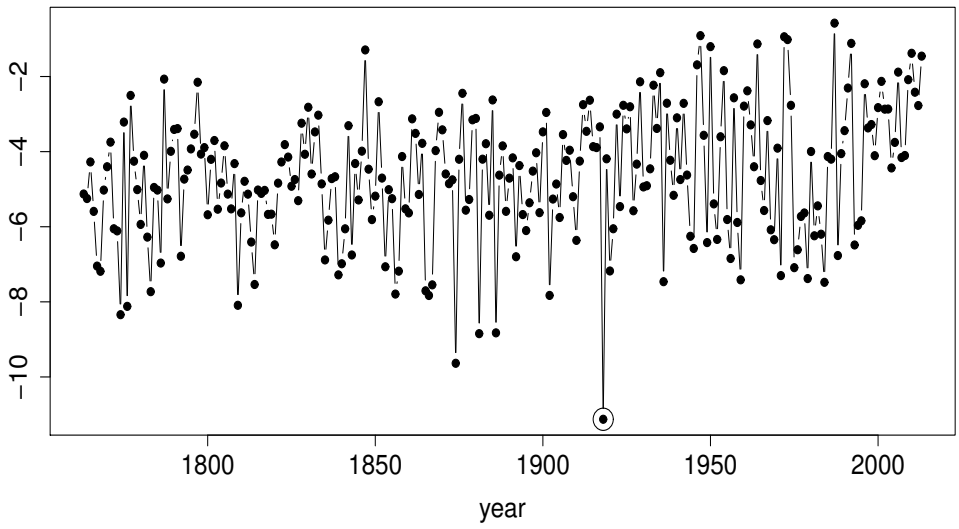


Fig. 4.1 January average temperature in Reykjavik recorded from 1763 to 2013.

with more details in Section 4.5, many existing change-point detection methods for the piecewise-linear model fail in this type of signal that includes abrupt local features or frequent change-points. In Section 4.5.1, we show that our new methodology can detect

not only change-points in linear trend but it can also detect the 1918 observation as a point anomaly, which the other methods do not achieve.

The change-point detection procedure proposed in this chapter is referred to as TrendSegment. It is designed to work well in detecting not only long trend segments and point anomalies but also short trend segments that are not necessarily classified as point anomalies. Later in this chapter, we show that TrendSegment offers good performance in estimating the number and locations of change-points across a wide range of signals containing a mix of constant and linear segments and/or point anomalies. TrendSegment is also shown to be statistically consistent and computationally efficient. Core to the TrendSegment procedure is a new Tail-Greedy Unbalanced Wavelet (TGUW) transform: a conditionally orthonormal, bottom-up transformation for univariate data sequences through an adaptively constructed unbalanced wavelet basis, which results in a sparse representation of the data. The TGUW transform, which underlies TrendSegment, is designed to handle scenarios involving frequent change-points or abrupt local features, in which many existing change-point detection methods fail as illustrated later in this chapter. It constructs a data-adaptive wavelet basis in a bottom-up way in that it consecutively merges neighbouring segments of the data from its finest level. This enables it to identify local features at an early stage before it proceeds to focus on more global features corresponding to longer data segments.

We emphasise that the TGUW transform is an extension of the Tail-Greedy Unbalanced Haar (TGUH, Fryzlewicz (2018b)) transform illustrated in Section 2.2.1, a bottom-up, agglomerative and data-adaptive transformation of univariate sequences that facilitates change-point detection in the "piecewise-constant" sequence model. The extension to the TGUW transform is done by constructing adaptive wavelets instead of adaptive Haar, which enables change-point detection in the "piecewise-linear" model. In principle, it can be extended to higher-order piecewise polynomials, but we

do not pursue this in the current work. We emphasise that this extension from TGUH to TGUW is both conceptually and technically non-trivial, due to the fact that it is not a priori clear how to construct a suitable wavelet basis in TGUW for wavelets other than adaptive Haar. In the TGUH transform, constructing the adaptive Haar wavelet is relatively simple; as formulated in (2.19), for any pair of neighbouring regions $[p, q]$ and $[q+1, r]$, the corresponding detail-type coefficient $d_{p,q,r}$ (whose magnitude represents the strength of the corresponding local constancy) is equal to the formulation of CUSUM statistic. Therefore, the corresponding wavelet function $\psi_{p,q,r}$ has a form of piecewise-constant function as follows:

$$\begin{aligned} d_{p,q,r} &= \sqrt{\frac{r-q}{r-p+1}} s_{p,q} - \sqrt{\frac{q-p+1}{r-p+1}} s_{q+1,r}, \\ &= \sqrt{\frac{r-q}{(r-p+1)(q-p+1)}} \sum_{t=p}^{q} X_t - \sqrt{\frac{q-p+1}{(r-p+1)(r-q)}} \sum_{t=q+1}^{r} X_t, \\ &= \langle \boldsymbol{X}, \psi_{p,q,r} \rangle, \end{aligned} \tag{4.2}$$

where $\boldsymbol{X} = \{X_1, X_2, \ldots, X_T\}^\top$ and $s_{p,q}$ is a smooth coefficient such that $s_{p,r} = (r-p+1)^{-1/2} \sum_{s=p}^{r} X_s$ which can be interpreted as a scaled local sample mean. This enables us to perform a local orthonormal transform for a pair of smooth coefficients, $(s_{p,q}, s_{q+1,r})$, through a unique orthonormal matrix which returns the new (smooth, detail) pair as follows:

$$\begin{pmatrix} s_{p,r} \\ d_{p,q,r} \end{pmatrix} = \begin{pmatrix} \sqrt{\frac{q-p+1}{r-p+1}} & \sqrt{\frac{r-q}{r-p+1}} \\ \sqrt{\frac{r-q}{r-p+1}} & -\sqrt{\frac{q-p+1}{r-p+1}} \end{pmatrix} \begin{pmatrix} s_{p,q} \\ s_{q+1,r} \end{pmatrix} = \Lambda_{p,q,r} \begin{pmatrix} s_{p,q} \\ s_{q+1,r} \end{pmatrix}, \tag{4.3}$$

where $\Lambda_{p,q,r}$ is an orthonormal matrix. However, this does not occur in TGUW; the corresponding local orthonormal transform matrix for performing each merge has the dimension 3×3 (instead of 2×2 in TGUH) and the matrix does not have uniqueness. The orthonormal matrix, Λ, for the local TGUW transform is composed of two low

filter vectors ($\boldsymbol{\ell}_1$ and $\boldsymbol{\ell}_2$) of length 3 (which correspond to two new smooth coefficients $s^1_{p,r}$, $s^2_{p,r}$) and one high filter vector $\boldsymbol{h}$ of length 3 (which corresponds to one new detail coefficient $d_{p,q,r}$) as follows:

$$\begin{pmatrix} s^1_{p,r} \\ s^2_{p,r} \\ d_{p,q,r} \end{pmatrix}_{3\times 1} = \begin{pmatrix} \boldsymbol{\ell}_1^\top \\ \boldsymbol{\ell}_2^\top \\ \boldsymbol{h}^\top \end{pmatrix}_{3\times 3} \begin{pmatrix} s_1 \\ s_2 \\ s_3 \end{pmatrix}_{3\times 1} = \Lambda_{3\times 3} \begin{pmatrix} s_1 \\ s_2 \\ s_3 \end{pmatrix}_{3\times 1} . \tag{4.4}$$

Unlike the TGUH transform in (4.3), the TGUW transform in (4.4) returns two smooth coefficients. Due to the non-uniqueness of those two low filter vectors in (4.4), it is not clear how to understand and interpret the corresponding two new smooth coefficients. The non-uniqueness itself does not affect the recursively performed orthonormal transformations, but we should impose a certain guiding principle in the way the merges are performed to guarantee that an adaptively constructed unbalanced wavelet basis has orthonormality. Establishing this guiding principle (which will be specified in Section 4.2.2) is new and the most challenging part in our unbalanced wavelet transform compared to the unbalanced Haar transform, by which our algorithm is able to detect changes in the linear trend and point anomalies at the same time. The computational cost of TGUW is the same as TGUH. Important properties of the TGUW transform include conditional orthonormality, nonlinearity and "tail-greediness", and will be investigated in Section 4.2. The TGUW transform is the first step of the TrendSegment procedure, which involves four steps.

The remainder of this chapter is organised as follows. Section 4.2 gives a full description of the TrendSegment procedure and the relevant theoretical results are presented in Section 4.3. The supporting simulation studies are described in Section 4.4 and our methodology is illustrated in Section 4.5 through climate datasets. The

proofs of our main theoretical results are in Section 4.6. The TrendSegment procedure is implemented in the R package `trendsegmentR`.

4.2 Methodology

4.2.1 Summary of TrendSegment

The TrendSegment procedure for estimating the number and the locations of change-points includes four steps. We give a broad picture first and outline details in later sections.

1. *TGUW transformation.* Perform the TGUW transform; a bottom-up unbalanced adaptive wavelet transformation of the input data $X_1, \ldots, X_T$ by recursively applying local conditionally orthonormal transformations. This produces a data-adaptive multiscale decomposition of the data with $T-2$ detail-type coefficients and 2 smooth coefficients. The resulting conditionally orthonormal transform of the data hopes to push most of the energy of the signal in only a few detail-type coefficients arising at coarse levels. This sparse representation of the data justifies thresholding in the next step.
2. *Thresholding.* Set to zero those detail coefficients whose magnitude is smaller than a pre-specified threshold as long as all the non-zero detail coefficients are connected to each other in the tree structure. This step performs "pruning" as a way of deciding the significance of the sparse representation obtained in step 1.
3. *Inverse TGUW transformation.* Obtain an initial estimate of f_t by carrying out the inverse TGUW transformation of the thresholded coefficient tree. The resulting estimator can be shown to be l_2-consistent, but not yet consistent for the number (N) and the locations of change-points $(\eta_1, \ldots, \eta_N)$.

4. *Post-processing.* Post-process the estimate from step 3 by removing some change-points perceived to be spurious, which enables us to achieve estimation consistency for N and $\eta_1, \ldots, \eta_N$.

We devote the following four sections to describing each step above in order.

4.2.2 TGUW transformation

Key principles of the TGUW transform

In the initial stage, the data are considered smooth coefficients and the TGUW transform iteratively updates the sequence of smooth coefficients by merging the adjacent sections of the data which are the most likely to belong to the same segment. The merging is done by performing an adaptively constructed orthonormal transformation to the chosen triplet of the smooth coefficients and in doing so, a data-adaptive unbalanced wavelet basis is established. The TGUW transform is completed after $T - 2$ such orthonormal transformations and each merge is performed under the following principles.

1. In each merge, three adjacent smooth coefficients are selected and the orthonormal transformation converts those three values into one detail and two (updated) smooth coefficients. The size of the detail coefficient gives information about the strength of the local linearity and the two updated smooth coefficients are associated with the estimated parameters (intercept and slope) of the local linear regression performed on the raw observations corresponding to the initially chosen three smooth coefficients.
2. *"Two together" rule.* The two smooth coefficients returned by the orthonormal transformation are paired in the sense that both contain information about one local linear regression fit. Thus, we require that any such pair of smooth coefficients cannot be separated in any subsequent merges. We refer to this recipe as the "two together" rule.

3. To decide which triplet of smooth coefficients should be merged next, we compare the corresponding detail coefficients as their magnitude represents the strength of the corresponding local linear trend; the smaller the (absolute) size of the detail, the smaller the local deviation from linearity. Smooth coefficients corresponding to the smallest detail coefficients have priority in merging.

As merging continues under the "two together" rule, all mergings can be classified into one of three forms, Type 1: merging three initial smooth coefficients, Type 2: merging one initial and a paired smooth coefficient and Type 3: merging two sets of (paired) smooth coefficients. Note that Type 3 is composed of two consecutive merges of triplets and more details are given later.

Table 4.1 Notation. See Section 4.2.2 for formulae for the terms listed.

X_p	p^{th} element of the observation vector $\boldsymbol{X} = \{X_1, X_2, \ldots, X_T\}^\top$.
$s^0_{p,p}$	p^{th} initial smooth coefficient of the vector $\boldsymbol{s}^0$ where $\boldsymbol{X} = \boldsymbol{s}^0$.
$d_{p,q,r}$	detail coefficient obtained from $\{X_p, \ldots, X_r\}$ (merges of Types 1 or 2).
$s^1_{p,r}, s^2_{p,r}$	smooth coefficients obtained from $\{X_p, \ldots, X_r\}$, paired under the "two together" rule.
$d^1_{p,q,r}, d^2_{p,q,r}$	paired detail coefficients obtained by merging two adjacent subintervals, $\{X_p, \ldots, X_q\}$ and $\{X_{q+1}, \ldots, X_r\}$, where $r > q+2$ and $q > p+1$ (merge of Type 3).
$\boldsymbol{s}$	data sequence vector containing the (recursively updated) smooth and detail coefficients from the initial input $\boldsymbol{s}^0$.

Example

We now provide a simple example of the TGUW transformation to help readers understand the entire procedure at a glance. The accompanying illustration is in Figure 4.2 and the notation for this example and for the general algorithm introduced later is in Table 4.1. This example shows single merges at each pass through the data. We will later generalise it to multiple passes through the data, which will speed up computation where the device is referred to as "tail-greediness". We refer

to j^{th} pass through the data as scale j. Assume that we have the initial input $\boldsymbol{s}^0 = (X_1, X_2, \ldots, X_8)$, so that the complete TGUW transform consists of 6 merges. We show 6 example merges one by one under the rules introduced above. This example demonstrates all three possible types of merges.

Scale $j = 1$. From the initial input $\boldsymbol{s}^0 = (X_1, \ldots, X_8)$, we consider 6 triplets (X_1, X_2, X_3), (X_2, X_3, X_4), (X_3, X_4, X_5), (X_4, X_5, X_6), (X_5, X_6, X_7), (X_6, X_7, X_8) and compute the size of the detail for each triplet, where the formula can be found in (4.5). Suppose that (X_2, X_3, X_4) gives the smallest size of detail, $|d_{2,3,4}|$, then merge (X_2, X_3, X_4) through the orthogonal transformation formulated in (4.7) and update the data sequence into $\boldsymbol{s} = (X_1, s^1_{2,4}, s^2_{2,4}, d_{2,3,4}, X_5, X_6, X_7, X_8)$. We categorise this transformation into Type 1 (merging three initial smooth coefficients).

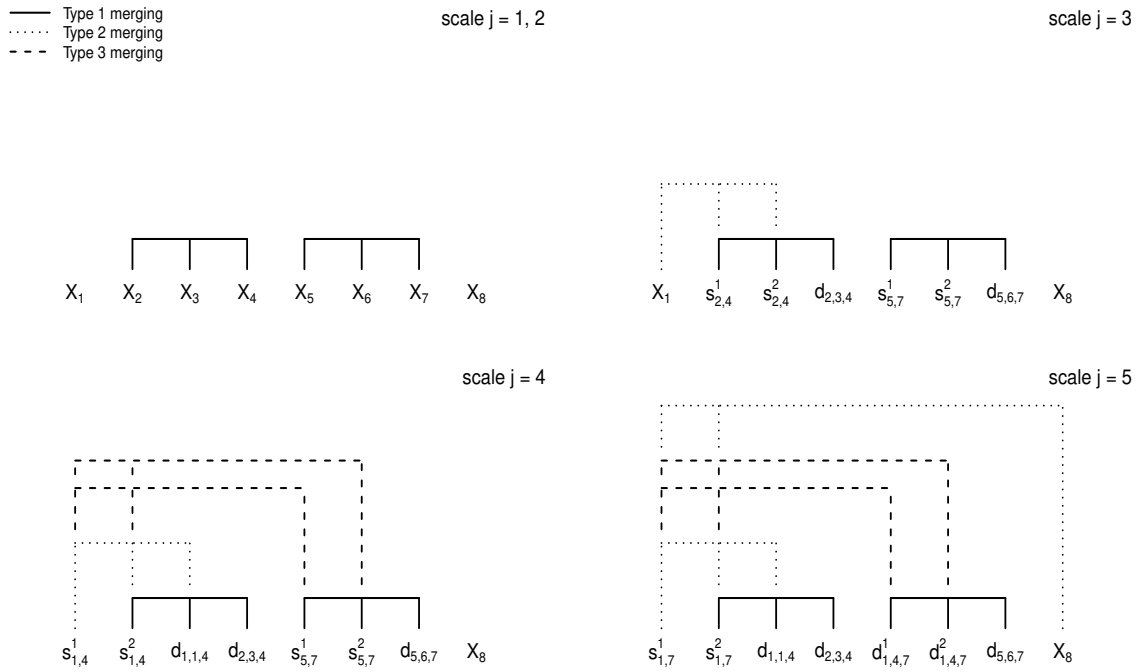


Fig. 4.2 Construction of tree for the example in Section 4.2.2; each diagram shows all merges performed up to the given scale.

Scale $j = 2$. From now on, the "two together" rule is applied. Ignoring any detail coefficients in $\boldsymbol{s}$, the possible triplets for next merging are $(X_1, s^1_{2,4}, s^2_{2,4})$, $(s^1_{2,4}, s^2_{2,4}, X_5)$,

(X_5, X_6, X_7), (X_6, X_7, X_8). We note that $(s^2_{2,4}, X_5, X_6)$ cannot be considered as a candidate for next merging under the "two together" rule as this triplet contains only one (not both) of the paired smooth coefficients returned by the previous merging. Assume that (X_5, X_6, X_7) gives the smallest size of detail coefficient $|d_{5,6,7}|$ among the four candidates, then we merge them through the orthogonal transformation formulated in (4.7) and now update the sequence into $\boldsymbol{s} = (X_1, s^1_{2,4}, s^2_{2,4}, d_{2,3,4}, s^1_{5,7}, s^2_{5,7}, d_{5,6,7}, X_8)$. This transformation is also Type 1.

Scale $j = 3$. We now compare four candidates for merging, $(X_1, s^1_{2,4}, s^2_{2,4})$, $(s^1_{2,4}, s^2_{2,4}, s^1_{5,7})$, $(s^2_{2,4}, s^1_{5,7}, s^2_{5,7})$ and $(s^1_{5,7}, s^2_{5,7}, X_8)$. The two triplets in middle, $(s^1_{2,4}, s^2_{2,4}, s^1_{5,7})$ and $(s^2_{2,4}, s^1_{5,7}, s^2_{5,7})$, are paired together as they contain two sets of paired smooth coefficients, $(s^1_{2,4}, s^2_{2,4})$ and $(s^1_{5,7}, s^2_{5,7})$, and if we were to treat these two triplets separately, we would be violating the "two together" rule. The summary detail coefficient for this pair of triplets is obtained as $d_{2,4,7} = \max(|d^1_{2,4,7}|, |d^2_{2,4,7}|)$, which is compared with those of the other triplets. Now suppose that $(X_1, s^1_{2,4}, s^2_{2,4})$ has the smallest size of detail; we merge this triplet and update the data sequence into $\boldsymbol{s} = (s^1_{1,4}, s^2_{1,4}, d_{1,1,4}, d_{2,3,4}, s^1_{5,7}, s^2_{5,7}, d_{5,6,7}, X_8)$. This transformation is of Type 2.

Scale $j = 4$. We now have two pairs of paired coefficients: $(s^1_{1,4}, s^2_{1,4})$ and $(s^1_{5,7}, s^2_{5,7})$. Therefore, with the "two together" rule in mind, the only possible options for merging are: to merge the two pairs into $(s^1_{1,4}, s^2_{1,4}, s^1_{5,7}, s^2_{5,7})$, or to merge $(s^1_{5,7}, s^2_{5,7})$ with X_8. Suppose that the first merging is preferred. The merge of $(s^1_{1,4}, s^2_{1,4})$ and $(s^1_{5,7}, s^2_{5,7})$ into $(s^1_{1,4}, s^2_{1,4}, s^1_{5,7}, s^2_{5,7})$ is of Type 3 and is performed in two stages as follows. In the first stage, we merge $(s^1_{1,4}, s^2_{1,4}, s^1_{5,7})$ and then update the sequence temporarily as $\boldsymbol{s} = (s^{1'}_{1,7}, s^{2'}_{1,7}, d_{1,1,4}, d_{2,3,4}, d^1_{1,4,7}, s^2_{5,7}, d_{5,6,7}, X_8)$. In the second stage, we merge $(s^{1'}_{1,7}, s^{2'}_{1,7}, s^2_{5,7})$, which gives the updated sequence $\boldsymbol{s} = (s^1_{1,7}, s^2_{1,7}, d_{1,1,4}, d_{2,3,4}, d^1_{1,4,7}, d^2_{1,4,7}, d_{5,6,7}, X_8)$. As a summary detail coefficients for this merge, we use $d_{1,4,7} = \max(|d^1_{1,4,7}|, |d^2_{1,4,7}|)$.

Scale $j = 5$. The only available triplet is now $(s^1_{1,7}, s^2_{1,7}, X_8)$, thus we perform this Type 2 merge and update the data sequence into $\boldsymbol{s} = (s^1_{1,8}, s^2_{1,8}, d_{1,1,4}, d_{2,3,4}, d^1_{1,4,7}, d^2_{1,4,7}, d_{5,6,7}, d_{1,7,8})$. The transformation is completed with the updated data sequence which contains $T - 2 = 6$ detail and 2 smooth coefficients.

TGUW transformation: general algorithm

We now formulate in generality the TGUW transformation illustrated in the above example. One of the important principles is "tail-greediness" (Fryzlewicz, 2018b) which enables us to reduce the computational complexity by performing multiple merges over non-overlapping regions in a single pass over the data. More specifically, it allows us to perform up to $\max\{2, \lceil \rho\alpha_j \rceil\}$ merges at each scale j, where α_j is the number of smooth coefficients in the data sequence $\boldsymbol{s}$ and $\rho \in (0, 1)$. The lower bound of 2 is essential to permit a Type 3 transformation, which consists of two merges.

Sometimes, we will be referring to a detail coefficient $d_{p,q,r}$ as $d^{(j,k)}_{p,q,r}$ or $d^{(j,k)}$, where $j = 1, \ldots, J$ is the scale of the transform (i.e. the consecutive pass through the data) at which $d_{p,q,r}$ was computed, $k = 1, \ldots, K(j)$ is the location index of $d_{p,q,r}$ within all scale j coefficients, and $d_{p,q,r}$ is $d^1_{p,q,r}$ or $d^2_{p,q,r}$ or $d_{p,q,r}$, depending on the type of merge. We now describe the TGUW algorithm.

1. At each scale j, find the set of triplets that are candidates for merging under the "two together" rule and compute the corresponding detail coefficients. Regardless of the type of merge, a detail coefficient $d_{p,q,r}$ is, in general, obtained as

$$d_{p,q,r} = a\boldsymbol{s}^1_{p:r} + b\boldsymbol{s}^2_{p:r} + c\boldsymbol{s}^3_{p:r}, \tag{4.5}$$

   where $p \le q < r$, $\boldsymbol{s}^k_{p:r}$ is the k^{th} smooth coefficient of the subvector $\boldsymbol{s}_{p:r}$ with a length of $r - p + 1$ and the constants a, b, c are the elements of the detail filter $\boldsymbol{h} = (a, b, c)^\top$. We note that (a, b, c) also depends on (p, q, r), but this is not reflected in the

notation, for simplicity. The detail filter is a weight vector used in computing the weighted sum of a triplet of smooth coefficients which should satisfy the condition that the detail coefficient is zero if and only if the corresponding raw observations over the merged regions have a perfect linear trend. If $(X_p, \ldots, X_r)$ are the raw observations associated with the triplet of the smooth coefficients $(\boldsymbol{s}^1_{p:r}, \boldsymbol{s}^2_{p:r}, \boldsymbol{s}^3_{p:r})$ under consideration, then the detail filter $\boldsymbol{h}$ is obtained in such a way as to produce zero detail coefficient only when $(X_p, \ldots, X_r)$ has a perfect linear trend, as the detail coefficient itself represents the extent of non-linearity in the corresponding region of data. This implies that the smaller the size of the detail coefficient, the closer the alignment of the corresponding data section with linearity. Specifically, the detail filter $\boldsymbol{h} = (a, b, c)^\top$ is established by solving the following equations,

$$\begin{aligned} a\boldsymbol{w}^{c,1}_{p:r} + b\boldsymbol{w}^{c,2}_{p:r} + c\boldsymbol{w}^{c,3}_{p:r} &= 0, \\ a\boldsymbol{w}^{l,1}_{p:r} + b\boldsymbol{w}^{l,2}_{p:r} + c\boldsymbol{w}^{l,3}_{p:r} &= 0, \\ a^2 + b^2 + c^2 &= 1, \end{aligned} \tag{4.6}$$

where $\boldsymbol{w}^{\cdot,k}_{p:r}$ is k^{th} non-zero element of the subvector $\boldsymbol{w}^{\cdot}_{p:r}$ with a length of $r - p + 1$, and $\boldsymbol{w}^c$ and $\boldsymbol{w}^l$ are weight vectors of constancy and linearity, respectively, in which the initial inputs have a form of $\boldsymbol{w}^c_0 = (1, 1, \ldots, 1)^\top, \boldsymbol{w}^l_0 = (1, 2, \ldots, T)^\top$. The last condition in (4.6) is to preserve the orthonormality of the transform. Intuitively, the detail filter $\boldsymbol{h}$ becomes a normal vector of the plane $\{(x, y, z) \mid x - 2y + z = 0\}$. The solution to (4.6) is unique up to multiplication by -1 and this can be simply shown by solving the equations e.g. $a + b + c = 0$, $a + 2b + 3c = 0$ and $a^2 + b^2 + c^2 = 1$.

2. Summarise all $d^{\cdot}_{p,q,r}$ constructed in step 1 to a (equal length or shorter) sequence of $d_{p,q,r}$ by finding a summary detail coefficient $d_{p,q,r} = \max(|d^1_{p,q,r}|, |d^2_{p,q,r}|)$ for any pair of detail coefficients constructed by type 3 merges.

3. Sort the size of the summarised detail coefficients $|d_{p,q,r}|$ obtained in step 2 in non-decreasing order.
4. Extract the (non-summarised) detail coefficient(s) $|d^{\cdot}_{p,q,r}|$ corresponding to the smallest (summarised) detail coefficient $|d_{p,q,r}|$ where both $|d^1_{p,q,r}|$ and $|d^2_{p,q,r}|$ should be extracted only if $d_{p,q,r} = \max(|d^1_{p,q,r}|, |d^2_{p,q,r}|)$. Repeat the extraction until $\max\{2, \lceil \rho\alpha_j \rceil\}$ (or all possible, whichever is the smaller number) detail coefficients have been obtained, as long as the region of the data corresponding to each detail coefficient extracted does not overlap with the regions corresponding to the detail coefficients already drawn.
5. For each $|d^{\cdot}_{p,q,r}|$ extracted in step 4, merge the corresponding smooth coefficients by updating the corresponding triplet in $\boldsymbol{s}$, $\boldsymbol{w}^c$ and $\boldsymbol{w}^l$ through the orthonormal transform as follows,

$$\begin{pmatrix} s^1_{p,r} \\ s^2_{p,r} \\ d^{\cdot}_{p,q,r} \end{pmatrix} = \begin{pmatrix} \boldsymbol{\ell}_1^\top \\ \boldsymbol{\ell}_2^\top \\ \boldsymbol{h}^\top \end{pmatrix} \begin{pmatrix} \boldsymbol{s}^1_{p:r} \\ \boldsymbol{s}^2_{p:r} \\ \boldsymbol{s}^3_{p:r} \end{pmatrix} = \Lambda \begin{pmatrix} \boldsymbol{s}^1_{p:r} \\ \boldsymbol{s}^2_{p:r} \\ \boldsymbol{s}^3_{p:r} \end{pmatrix}, \tag{4.7}$$

$$\begin{pmatrix} w^{c,1}_{p,r} \\ w^{c,2}_{p,r} \\ 0 \end{pmatrix} = \Lambda \begin{pmatrix} \boldsymbol{w}^{c,1}_{p:r} \\ \boldsymbol{w}^{c,2}_{p:r} \\ \boldsymbol{w}^{c,3}_{p:r} \end{pmatrix}, \quad \begin{pmatrix} w^{l,1}_{p,r} \\ w^{l,2}_{p,r} \\ 0 \end{pmatrix} = \Lambda \begin{pmatrix} \boldsymbol{w}^{l,1}_{p:r} \\ \boldsymbol{w}^{l,2}_{p:r} \\ \boldsymbol{w}^{l,3}_{p:r} \end{pmatrix}. \tag{4.8}$$

The key step is finding the 3×3 orthonormal matrix, Λ, which is composed of one detail and two low-pass filter vectors in its rows. Firstly the detail filter $\boldsymbol{h}^\top$ is determined to satisfy the conditions in (4.6), and then the two low-pass filters $(\boldsymbol{\ell}_1^\top, \boldsymbol{\ell}_2^\top)$ are obtained by satisfying the orthonormality of Λ. There is no uniqueness in the choice of $(\boldsymbol{\ell}_1^\top, \boldsymbol{\ell}_2^\top)$, but this has no effect on the orthonormal transformation itself. The details of this mechanism can be found in Section 4.2.6.

6. Go to step 1 and repeat at new scale $j = j + 1$ as long as we have at least three smooth coefficients in the updated data sequence $\boldsymbol{s}$.

More specifically, the detail coefficient in (4.5) is formulated for each type of merging introduced in Section 4.2.2 as follows.

Type 1: merging three initial smooth coefficients $(s^0_{p,p}, s^0_{p+1,p+1}, s^0_{p+2,p+2})$,

$$d_{p,p+1,p+2} = a_{p,p+1,p+2}s^0_{p,p} + b_{p,p+1,p+2}s^0_{p+1,p+1} + c_{p,p+1,p+2}s^0_{p+2,p+2}. \tag{4.9}$$

Type 2: merging one initial and a paired smooth coefficient $(s^0_{p,p}, s^1_{p+1,r}, s^2_{p+1,r})$,

$$d_{p,p,r} = a_{p,p,r}s^0_{p,p} + b_{p,p,r}s^1_{p+1,r} + c_{p,p,r}s^2_{p+1,r}, \quad \text{where} \quad p+2 < r, \tag{4.10}$$

similarly, when merging a paired smooth coefficient and one initial, $(s^1_{p,r-1}, s^2_{p,r-1}, s^0_{r,r})$,

$$d_{p,r-1,r} = a_{p,r-1,r}s^1_{p,r-1} + b_{p,r-1,r}s^2_{p,r-1} + c_{p,r-1,r}s^0_{r,r}, \quad \text{where} \quad p+2 < r. \tag{4.11}$$

Type 3: merging two sets of (paired) smooth coefficients, $(s^1_{p,q}, s^2_{p,q})$ and $(s^1_{q+1,r}, s^2_{q+1,r})$,

$$\begin{aligned} d^1_{p,q,r} &= a^1_{p,q,r}s^1_{p,q} + b^1_{p,q,r}s^2_{p,q} + c^1_{p,q,r}s^1_{q+1,r} \\ d^2_{p,q,r} &= a^2_{p,q,r}s^{01}_{p,r} + b^2_{p,q,r}s^{02}_{p,r} + c^2_{p,q,r}s^2_{q+1,r} \end{aligned} \quad \Longrightarrow \quad d_{p,q,r} = \max(|d^1_{p,q,r}|, |d^2_{p,q,r}|), \tag{4.12}$$

where $q > p+1$ and $r > q+2$. Importantly, the two consecutive merges in (4.12) are achieved by visiting the same two adjacent data regions twice. In this case, after the first detail coefficient, $d^1_{p,q,r}$, has been obtained, we instantly update the corresponding triplets $\boldsymbol{s}$, $\boldsymbol{w}^c$ and $\boldsymbol{w}^l$ via an orthonormal transform as defined in (4.7) and (4.8). Therefore, the second detail filter, $(a^2_{p,q,r}, b^2_{p,q,r}, c^2_{p,q,r})$, is constructed with the updated $\boldsymbol{w}^c$ and $\boldsymbol{w}^l$ in a way that satisfies the conditions (4.6).

The TGUW transform eventually converts the input data sequence $\boldsymbol{X}$ of length T into the sequence containing 2 smooth and $T-2$ detail coefficients through $T-2$ orthonormal transforms. The detail coefficients $d^{(j,k)}$ can be regarded as scalar products between $\boldsymbol{X}$ and a particular unbalanced wavelet basis $\psi^{(j,k)}$, where the formal representation is given as $\{d^{(j,k)} = \langle \boldsymbol{X}, \psi^{(j,k)} \rangle_{,j=1,\ldots,J,k=1,\ \ldots,K(j)}\}$ for detail coefficients and $s^1_{1,T} = \langle \boldsymbol{X}, \psi^{(0,1)} \rangle$, $s^2_{1,T} = \langle \boldsymbol{X}, \psi^{(0,2)} \rangle$ for the two smooth coefficients. The set $\{\psi^{(j,k)}\}$ is an orthonormal unbalanced wavelet basis for $\mathbb{R}^T$.

4.2.3 Thresholding

As the TGUW transform is performed in a way to push the l_2 energy of the input data to a small number of detail coefficients, the bulk of variability (= deviation from linearity) of the signal tends to be mainly captured by few detail coefficients computed at the later stages of the transform. This sparse representation of the input data justifies thresholding as a way of deciding the significance of each detail coefficient.

Two important rules of thresholding are referred to as the "connected" rule and the "two together" rule which should simultaneously be satisfied. These two rules are illustrated in Figure 4.3 by using the tree established in the example of Section 4.2.2. The diagram in top-row describes the "connected" rule which prunes the branches of the TGUW detail coefficients if and only if the detail coefficient itself and all of its children coefficients fall below a certain threshold in absolute value. If both $d_{1,1,4}$ and $d_{1,7,8}$ were to survive the initial thresholding, the "connected" rule would mean we also had to keep $d^1_{1,4,7}$ and $d^2_{1,4,7}$, which are the children of $d_{1,7,8}$ and the parents of $d_{1,1,4}$ in the TGUW coefficient tree.

The "two together" rule in thresholding is similar to the one in the TGUW transformation except it targets pairs of detail rather than smooth coefficients. It only applies to pairs of detail coefficients arising from Type 3 merges e.g. $(d^1_{1,4,7}, d^2_{1,4,7})$ in

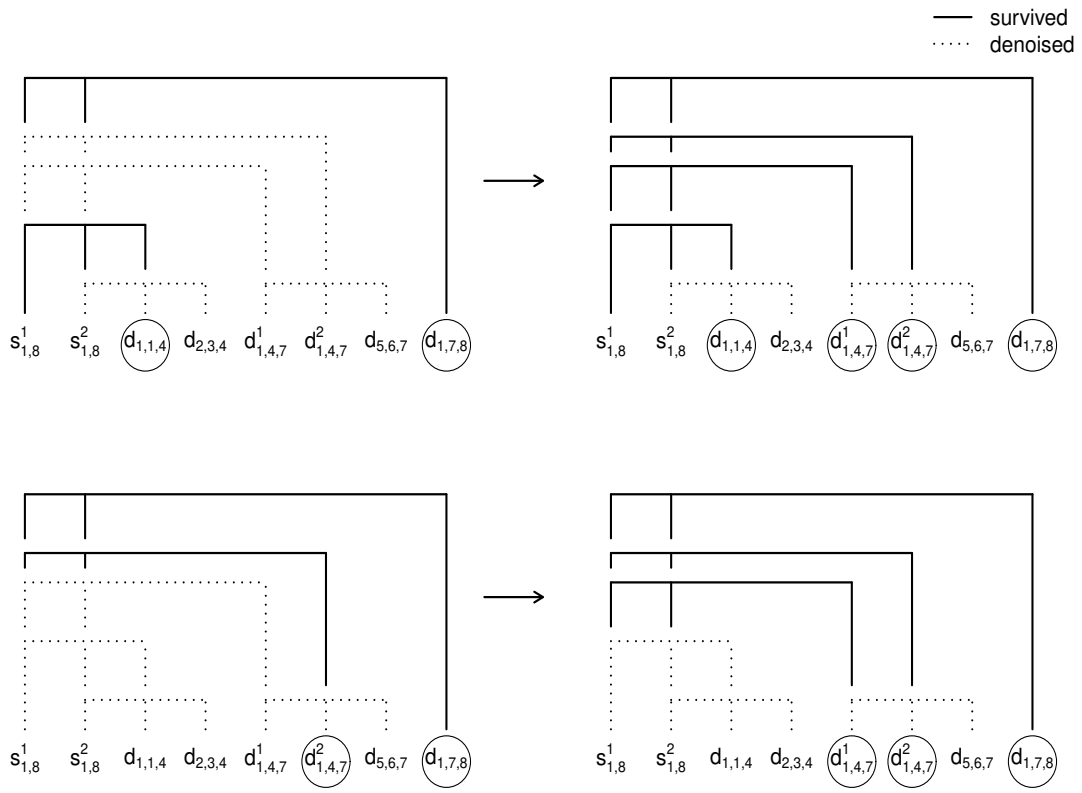


Fig. 4.3 The tree of mergings in the example illustrated in Section 4.2.2. Diagrams in left-column show the examples of tree obtained from initial hard thresholding, the one in top-right shows the tree after applying the "connected" rule and the one in bottom-right shows the tree after applying the "two together" rule described in Section 4.2.3. Solid line represents a survived merging and dashed line shows a denoised one.

bottom-row of Figure 4.3, in such a way that both detail coefficients should be kept if at least one survives the initial thresholding. This is a natural requirement as a pair of Type 3 detail coefficients effectively corresponds to a single merge of two adjacent regions.

Through the thresholding, we wish to estimate the underlying signal $\boldsymbol{f} = (f_1, \cdots, f_T)^\top$ in (4.1) by estimating $\mu^{(j,k)} = \langle \boldsymbol{f}, \psi^{(j,k)} \rangle$ where $\psi^{(j,k)}$ is an orthonormal unbalanced wavelet basis constructed in the TGUW transform from the data. Throughout the entire thresholding procedure, the "connected" and "two together" rules are applied in this order. We firstly apply the "connected" rule which gives us $\hat{\mu}_0^{(j,k)}$, the initial

estimator of $\mu^{(j,k)}$, as

$$\hat{\mu}_0^{(j,k)} = d_{p,q,r}^{(j,k)} \cdot \mathbb{I}\left\{ \exists (j',k') \in \mathcal{C}_{j,k} \quad \left| d_{p',q',r'}^{(j',k')} \right| > \lambda \right\}, \tag{4.13}$$

where $\mathbb{I}$ is an indicator function and

$$\mathcal{C}_{j,k} = \left\{ (j',k'), j' = 1,\ldots,j, k' = 1,\ldots,K(j') : d_{p',q',r'}^{(j',k')} \text{ is such that } [p',r'] \subseteq [p,r] \right\}. \tag{4.14}$$

Now the "two together" rule is applied to the initial estimator $\hat{\mu}_0^{(j,k)}$ to obtain the final estimator $\hat{\mu}^{(j,k)}$. We firstly note that two detail coefficients, $d_{p,q,r}^{(j,k)}$ and $d_{p',q',r'}^{(j',k')}$ are called "paired" when they are formed by Type 3 merges and when $(j,p,q,r) = (j',p',q',r')$. The 'two together" rule is formulated as below,

$$\hat{\mu}^{(j,k)} = \begin{cases} \hat{\mu}_0^{(j,k)}, & \text{if } d_{p,q,r}^{(j,k)} \text{ is not paired,} \\ \hat{\mu}_0^{(j,k)}, & \text{if } d_{p,q,r}^{(j,k)} \text{ is paired with } d_{p,q,r}^{(j,k')} \text{ and both } \hat{\mu}_0^{(j,k)} \text{ and } \hat{\mu}_0^{(j,k')} \text{ are} \\ & \text{zero or non-zero,} \\ d^{(j,k)}, & \text{if } d_{p,q,r}^{(j,k)} \text{ is paired with } d_{p,q,r}^{(j,k')} \text{ and } \hat{\mu}_0^{(j,k')} \neq 0 \text{ and } \hat{\mu}_0^{(j,k)} = 0. \end{cases} \tag{4.15}$$

These two rules are useful in that they not only produce a simpler shape of tree which is easier to prune but also give a more interpretable form of the estimated function $\tilde{\boldsymbol{f}}$ which will be produced later by the inverse TGUW transformation in Section 4.2.4. Only when the thresholding is performed in a way to satisfy both of the two rules introduced above, $\tilde{\boldsymbol{f}}$ is equivalent to the piecewise-linear function composed of best linear fits (in the least-squares sense) for each interval of linearity i.e. $\tilde{\boldsymbol{f}}$ is constructed as follows:

$$\tilde{f}_t = \tilde{\theta}_{\ell,1} + \tilde{\theta}_{\ell,2}\, t \quad \text{for} \quad t \in \left[\tilde{\eta}_{\ell-1} + 1, \tilde{\eta}_\ell \right], \quad \ell = 1,\ldots,\tilde{N}, \tag{4.16}$$

where $\tilde{\eta}_0 = 0$, $\tilde{\eta}_{\tilde{N}+1} = T$ and $(\tilde{\theta}_{\ell,1}, \tilde{\theta}_{\ell,2})$ are the OLS intercept and slope coefficients, respectively, for the corresponding pairs $\{(t, X_t),\ t \in [\tilde{\eta}_{\ell-1} + 1, \tilde{\eta}_\ell]\}$.

As an aside, we note that the number of survived detail coefficients are not exactly matched with the number of estimated change-points in $\tilde{\boldsymbol{f}}$ as a pair of detail coefficients arising from Type 3 merge are associated with a single change-point.

4.2.4 Inverse TGUW transformation

The estimator $\tilde{\boldsymbol{f}}$ of the true signal $\boldsymbol{f}$ in (4.1) is obtained by inverting the orthonormal transformation in (4.7) in reverse order to that in which they were originally performed. This inverse TGUW transformation is referred to as TGUW^{-1}, and thus

$$\tilde{\boldsymbol{f}} = \text{TGUW}^{-1}\Big\{ \hat{\mu}^{(j,k)}, j = 1, \ldots, J,\ k = 1, \ldots, K(j) \parallel s^1_{1,T}, s^2_{1,T} \Big\}, \tag{4.17}$$

where $\hat{\mu}^{(j,k)}$ is the sequence of the thresholded detail coefficients in (4.15) and $\parallel$ denotes vector concatenation. The inverse TGUW transform can be illustrated by borrowing the simplified notation in (4.4) as follows:

$$\begin{pmatrix} s_1 \\ s_2 \\ s_3 \end{pmatrix} = \begin{pmatrix} \boldsymbol{\ell}_1^\top \\ \boldsymbol{\ell}_2^\top \\ \boldsymbol{h}^\top \end{pmatrix}^{-1} \begin{pmatrix} s^1 \\ s^2 \\ d \end{pmatrix} = \Lambda^\top \begin{pmatrix} s^1 \\ s^2 \\ d \end{pmatrix} = \begin{pmatrix} \boldsymbol{\ell}_1 & \boldsymbol{\ell}_2 & \boldsymbol{h} \end{pmatrix} \begin{pmatrix} s^1 \\ s^2 \\ d \end{pmatrix}, \tag{4.18}$$

where the transform matrix Λ is orthogonal such that $\Lambda^\top = \Lambda^{-1}$.

4.2.5 Post-processing for consistency of change-point detection

As will be specified in Theorem 4.1 of Section 4.3, the piecewise-linear estimator $\tilde{\boldsymbol{f}}$ in (4.17) possibly overestimates the number of change-points. This is because $\tilde{\boldsymbol{f}}$ is l_2

consistent (i.e. $T^{-1}\sum_{i=1}^{T}(\tilde{f}_i - f_i)^2$ converges to zero with probability approaching to 1 as $T \to \infty$), but not yet consistent for the number and the locations of change-points. Lin et al. (2017) show that we can usually post-process l_2-consistent estimators as a fast enough l_2 error rate implies that each true change-point has an estimator nearby. To remove the spurious estimated change-points and to achieve the consistency of the number and the locations of the estimated change-points, we borrow the post-processing framework of Fryzlewicz (2018b). The post-processing methodology includes two stages, i) execution of three steps, TGUW transform, thresholding and inverse TGUW transform, again to the estimator $\tilde{\boldsymbol{f}}$ in (4.17) and ii) examination of regions containing only one estimated change-point to check for its significance. As will be described below, these two stages of post-processing are not used in practice (thus ignored in simulations and data analysis) for some practical reasons, however they are essential to achieve the consistency of the number and the locations of the estimated change-points as shown in Theorem 4.3.

Stage 1.

We transform the estimated function $\tilde{\boldsymbol{f}}$ in (4.17) with change-points $(\tilde{\eta}_1, \tilde{\eta}_2, \ldots, \tilde{\eta}_{\tilde{N}})$ into a new estimator $\tilde{\tilde{\boldsymbol{f}}}$ with corresponding change-points $(\tilde{\tilde{\eta}}_1, \tilde{\tilde{\eta}}_2, \ldots, \tilde{\tilde{\eta}}_{\tilde{\tilde{N}}})$. Using $\tilde{\boldsymbol{f}}$ in (4.17) as an input data sequence $\boldsymbol{s}$, we perform the TGUW transform as presented in Section 4.2.2, but in a greedy rather than tail-greedy way such that only one detail coefficient $d^{(j,1)}$ is produced at each scale j, and thus $K(j) = 1$ for all j. We repeat to produce detail coefficients until the first detail coefficient such that $|d^{(j,1)}| > \lambda$ is obtained where λ is the parameter used in the thresholding procedure described in Section 4.2.3. Once the condition, $|d^{(j,1)}| > \lambda$, is satisfied, stop merging and relabel the surviving change-points as $(\tilde{\tilde{\eta}}_1, \tilde{\tilde{\eta}}_2, \ldots, \tilde{\tilde{\eta}}_{\tilde{\tilde{N}}})$ and construct the new estimator $\tilde{\tilde{\boldsymbol{f}}}$ as

$$\tilde{\tilde{f}}_t = \tilde{\tilde{\theta}}_{\ell,1} + \tilde{\tilde{\theta}}_{\ell,2}\, t \quad \text{for} \quad t \in \left[\tilde{\tilde{\eta}}_{\ell-1} + 1, \tilde{\tilde{\eta}}_{\ell}\right], \quad \ell = 1, \ldots, \tilde{\tilde{N}}, \tag{4.19}$$

where $\tilde{\tilde{\eta}}_0 = 0$, $\tilde{\tilde{\eta}}_{\tilde{\tilde{N}}+1} = T$ and $(\tilde{\tilde{\theta}}_{\ell,1}, \tilde{\tilde{\theta}}_{\ell,2})$ are the OLS intercept and slope coefficients, respectively, for the corresponding pairs $\{(t, X_t),\ t \in \left[\tilde{\tilde{\eta}}_{\ell-1} + 1, \tilde{\tilde{\eta}}_{\ell}\right]\}$. The exception is when the region under consideration only contains a single data point X_{t_0} (a situation we refer to as a point anomaly throughout the chapter), in which case fitting a linear regression is impossible. We then set $\tilde{\tilde{f}}_{t_0} = X_{t_0}$.

Stage 2.

From the estimator $\tilde{\tilde{f}}_t$ in Stage 1, we obtain the final estimator $\hat{\boldsymbol{f}}$ by pruning the change-points $(\tilde{\tilde{\eta}}_1, \tilde{\tilde{\eta}}_2, \ldots, \tilde{\tilde{\eta}}_{\tilde{\tilde{N}}})$ in $\tilde{\tilde{f}}_t$. For each $i = 1, \ldots, \tilde{\tilde{N}}$, compute the corresponding detail coefficient d_{p_i,q_i,r_i} as described in (4.10)-(4.12), where $p_i = \left\lfloor \frac{\tilde{\tilde{\eta}}_{i-1} + \tilde{\tilde{\eta}}_i}{2} \right\rfloor + 1$, $q_i = \tilde{\tilde{\eta}}_i$ and $r_i = \left\lceil \frac{\tilde{\tilde{\eta}}_i + \tilde{\tilde{\eta}}_{i+1}}{2} \right\rceil$. Now prune by finding the minimiser $\ell_0 = \arg\min_i |d_{p_i,q_i,r_i}|$ and removing $\tilde{\tilde{\eta}}_{\ell_0}$ and setting $\tilde{\tilde{N}} := \tilde{\tilde{N}} - 1$ if $|d_{p_{\ell_0},q_{\ell_0},r_{\ell_0}}| \leq \lambda$ where λ is the same as in Section 4.2.3. Then relabel the change-points with the subscripts $i = 1, \ldots, \tilde{\tilde{N}}$ under the convention $\tilde{\tilde{\eta}}_0 = 0$, $\tilde{\tilde{\eta}}_{\tilde{\tilde{N}}+1} = T$. Repeat the pruning while we can find ℓ_0 which satisfies the condition $\left|d_{p_{\ell_0},q_{\ell_0},r_{\ell_0}}\right| < \lambda$. Otherwise, stop, set $\hat{N}$ as the number of detected change-points and reconstruct the change-points $\hat{\eta}_i$ in increasing order for $\ell = 0, \ldots, \hat{N} + 1$ where $\hat{\eta}_0 = 0$ and $\hat{\eta}_{\hat{N}+1} = T$. The estimated function $\hat{\boldsymbol{f}}$ is obtained by simple linear regression for each region determined by the final change-points $\hat{\eta}_1, \ldots, \hat{\eta}_{\hat{N}}$ as in (4.19), with the exception for point anomalies as described in Stage 1 above.

Through these two stages of post-processing, the estimation of the number and the locations of change-points becomes consistent, and the relevant theoretical results can be found in Section 4.3. Based on our empirical experience, Stage 1 rarely makes a difference in practice but causes an additional computational cost, and Stage 2 tends to over-prune change-points estimates and makes the procedure computationally heavy. For these practical reasons, in what follows, we recommend to use $\tilde{\boldsymbol{f}}$ in (4.17) as the

estimator of the TrendSegment procedure and disable Stages 1 and 2 of post-processing by default.

4.2.6 Extra discussion of TGUW transformation

Sparse representation.

We first recall that the TGUW transformation is obtained by a data-adaptively chosen orthonormal basis in $\mathbb{R}^T$ as follows,

$$\begin{pmatrix} s^1_{1,T} \\ s^2_{1,T} \\ \begin{pmatrix} d^{(j,k)}{}_{j=1,\ldots,J,k=1,\ldots,K(j)} \end{pmatrix} \end{pmatrix} = \begin{pmatrix} \psi^{(0,1)} \\ \psi^{(0,2)} \\ \begin{pmatrix} \psi^{(j,k)}{}_{j=1,\ldots,J,k=1,\ldots,K(j)} \end{pmatrix} \end{pmatrix} \begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_T \end{pmatrix} = \Psi_{T\times T} \begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_T \end{pmatrix}, \tag{4.20}$$

where Ψ is an orthogonal matrix. The orthogonal transformation matrix Ψ in (4.20) contains T orthonormal basis vectors in its rows that can be categorised into two: 1) $\left\{\psi^{(j,k)}{}_{j=1,\ldots,J,k=1,\ldots,K(j)}\right\}$ corresponding to detail coefficients $\left\{d^{(j,k)}{}_{j=1,\ldots,J,k=1,\ldots,K(j)}\right\}$ where $d^{(j,k)} = \langle \boldsymbol{X}, \psi^{(j,k)} \rangle$ and 2) $\psi^{(0,1)}$ and $\psi^{(0,2)}$ corresponding to two smooth coefficients, $s^1_{1,T} = \langle \boldsymbol{X}, \psi^{(0,1)} \rangle$ and $s^2_{1,T} = \langle \boldsymbol{X}, \psi^{(0,2)} \rangle$.

The TGUW transform is linear and orthonormal only when conditioning on the order in which the merges are performed. The orthonormality of the unbalanced wavelet basis, $\{\psi^{(j,k)}\}$, implies Parseval's identity:

$$\sum_{t=1}^{T} X_t^2 = \sum_{j=1}^{J} \sum_{k=1}^{K(j)} (d^{(j,k)})^2 + (s^1_{1,T})^2 + (s^2_{1,T})^2. \tag{4.21}$$

Furthermore, the filters $(\psi^{(0,1)}, \psi^{(0,2)})$ corresponding to the two smooth coefficients $s^1_{1,T}$ and $s^2_{1,T}$ form an orthonormal basis of the subspace $\{(x_1, x_2, \ldots, x_T) \mid x_1 - x_2 = x_2 - x_3 = \cdots = x_{T-1} - x_T\}$ of $\mathbb{R}^T$. (See later part of this section for further details.)

This implies

$$\sum_{t=1}^{T} X_t^2 - (s_{1,T}^1)^2 - (s_{1,T}^2)^2 = \sum_{t=1}^{T} (X_t - \hat{X}_t)^2 \tag{4.22}$$

where $\hat{\boldsymbol{X}} = s_{1,T}^1 \psi^{(0,1)} + s_{1,T}^2 \psi^{(0,2)}$ is the best linear regression fit to $\boldsymbol{X}$ achieved by minimising the sum of squared errors. The equation (4.22) can be simply shown as follows:

$$\begin{aligned}
\sum_{t=1}^{T} (X_t - \hat{X}_t)^2 =& (\boldsymbol{X} - s_{1,T}^1 \psi^{(0,1)} - s_{1,T}^2 \psi^{(0,2)})^\top (\boldsymbol{X} - s_{1,T}^1 \psi^{(0,1)} - s_{1,T}^2 \psi^{(0,2)}) \\
=& \boldsymbol{X}^\top \boldsymbol{X} + (s_{1,T}^1)^2 + (s_{1,T}^2)^2 - 2 s_{1,T}^1 \langle \boldsymbol{X}, \psi^{(0,1)} \rangle - 2 s_{1,T}^2 \langle \boldsymbol{X}, \psi^{(0,2)} \rangle \\
=& \boldsymbol{X}^\top \boldsymbol{X} - (s_{1,T}^1)^2 - (s_{1,T}^2)^2.
\end{aligned}$$

This, combined with the Parseval's identity above, implies,

$$\sum_{t=1}^{T} (X_t - \hat{X}_t)^2 = \sum_{j=1}^{J} \sum_{k=1}^{K(j)} (d^{(j,k)})^2. \tag{4.23}$$

By construction, the detail coefficients $|d^{(j,k)}|$ obtained in the initial stages of the TGUW transform tend to be small in magnitude. Then the Parseval's identity in (4.21) implies that a large portion of $\sum_{t=1}^{T} (X_t - \hat{X}_t)^2$ is explained by only a few large $|d^{(j,k)}|$'s arising in the later stages of the transform; in this sense, the TGUW transform provides sparsity of signal representation.

Computational complexity.

Assume that α_j smooth coefficients are available in the data sequence $\boldsymbol{s}$ at scale j and we allow the algorithm to merge up to $\lceil \rho \alpha_j \rceil$ many triplets (unless their corresponding data regions overlap) where $\rho \in (0, 1)$ is a constant. This gives us at most $(1-\rho)^j T$ smooth coefficients remaining in $\boldsymbol{s}$ after j scales. Solving for $(1-\rho)^j T \leq 2$ gives the largest number of scales J as $\left\lceil \log(T)/\log\left((1-\rho)^{-1}\right) + \log(2)/\log(1-\rho) \right\rceil$, at which point the

TGUW transform terminates with two smooth coefficients remaining. Considering that the most expensive step at each scale is sorting which takes $O(T\log(T))$ operations, the computational complexity of the TGUW transformation is $O(T\log^2(T))$.

Shape of the unbalanced wavelet basis.

We now explore the shape of the adaptively constructed unbalanced wavelet basis. First, we denote that $\psi^{(j,k)}$ in (4.20) is sometimes referred to as $\psi^{(j,k)}_{p,q,r}$. One of the important properties of the unbalanced wavelet basis is that $\psi^{(j,k)}_{p,q,r}$ always has a shape of linear trend in regions that are previously merged and this linearity will also be preserved in future merges, as long as later transforms are performed under the "two together" rule. For example, as mentioned earlier in this section, two vectors, $(\psi^{(0,1)}, \psi^{(0,2)})$, corresponding to the two smooth coefficients $s^1_{1,T}$ and $s^2_{1,T}$, have linear trends in the region $[1, T]$ as they form an orthonormal basis of the subspace $\{(x_1, x_2, \ldots, x_T) \mid x_1 - x_2 = x_2 - x_3 = \cdots = x_{T-1} - x_T\}$ of $\mathbb{R}^T$. This is due to the fact that the local orthonormal transforms continue in a way of extending the geometric dimension of subspace in which an orthonormal basis lives.

Through an illustrative example, we now show how a basis vector $\psi^{(j,k)}_{p,q,r}$ keeps its linearity in subregions that are already merged in previous scales, which includes a geometric interpretation of the TGUW transformation. Suppose that the initial data sequence is $\boldsymbol{s}^0 = (X_1, \ldots, X_5)$ and the initial weight vectors of constancy and linearity are $\boldsymbol{w}^c_0 = (1, 1, 1, 1, 1)^\top$ and $\boldsymbol{w}^l_0 = (1, 2, 3, 4, 5)^\top$, respectively. As we have the data sequence of length 5, the complete TGUW transform consists of 3 orthonormal transformations and the most important task for each transform is finding an appropriate orthonormal matrix.

First merge. Assume that (X_3, X_4, X_5) is chosen as the first triplet to be merged. To find the values of the transform matrix Λ,

$$\Lambda = \begin{pmatrix} \ell_{1,1} & \ell_{1,2} & \ell_{1,3} \\ \ell_{2,1} & \ell_{2,2} & \ell_{2,3} \\ a & b & c \end{pmatrix} = \begin{pmatrix} \boldsymbol{\ell}_1^\top \\ \boldsymbol{\ell}_2^\top \\ \boldsymbol{h}^\top \end{pmatrix}, \tag{4.24}$$

we first seek the detail filter, $\boldsymbol{h}$, which satisfies the conditions (1) $\boldsymbol{h}^\top \boldsymbol{w}^c_{0,3:5} = 0$, (2) $\boldsymbol{h}^\top \boldsymbol{w}^l_{0,3:5} = 0$ and (3) $\boldsymbol{h}^\top \boldsymbol{h} = 1$, where $\boldsymbol{w}^{\cdot}_{0,p:r}$ is the subvector of length $r-p+1$. Thus, $\boldsymbol{h}$ is obtained as a normal vector to the plane $\{(x,y,z) \mid x-2y+z=0\}$. Then, two low filter vectors ($\boldsymbol{\ell}_1$ and $\boldsymbol{\ell}_2$) are obtained under the conditions, (1) $\boldsymbol{\ell}_1^\top \boldsymbol{h} = 0$, (2) $\boldsymbol{\ell}_2^\top \boldsymbol{h} = 0$, (3) $\boldsymbol{\ell}_1^\top \boldsymbol{\ell}_2 = 0$ and (4) $\boldsymbol{\ell}_1^\top \boldsymbol{\ell}_1 = \boldsymbol{\ell}_2^\top \boldsymbol{\ell}_2 = 1$ which implies that $\boldsymbol{\ell}_1$ and $\boldsymbol{\ell}_2$ form an arbitrary orthonormal basis of the plane $\{(x,y,z) \mid x-2y+z=0\}$ and this guarantees the linear trend of $\boldsymbol{\ell}_1$ and $\boldsymbol{\ell}_2$. Now, the orthonormal transform updates the data sequence and weight vectors as follows,

$$\begin{aligned} \boldsymbol{s}^0 = (X_1, \ldots, X_5) \quad &\rightarrow \quad \boldsymbol{s} = (X_1, X_2, s^1_{3,5}, s^2_{3,5}, d_{3,4,5}), \\ \boldsymbol{w}^c_0 = (1,1,1,1,1)^\top \quad &\rightarrow \quad \boldsymbol{w}^c = (1,1,e_{c_1}, e_{c_2}, 0)^\top, \\ \boldsymbol{w}^l_0 = (1,2,3,4,5)^\top \quad &\rightarrow \quad \boldsymbol{w}^l = (1,2,e_{l_1}, e_{l_2}, 0)^\top, \end{aligned} \tag{4.25}$$

where the constants (e_{c_1}, e_{c_2}) and (e_{l_1}, e_{l_2}) are obtained by $\Lambda \boldsymbol{w}^c_{0,3:5} = (e_{c_1}, e_{c_2}, 0)^\top$ and $\Lambda \boldsymbol{w}^l_{0,3:5} = (e_{l_1}, e_{l_2}, 0)^\top$, respectively. As $\boldsymbol{\ell}_1$ and $\boldsymbol{\ell}_2$ form an orthonormal basis of the plane $\{(x,y,z) \mid x-2y+z=0\}$, e_{c_1}, e_{c_2} and e_{l_1}, e_{l_2} are unique constants which represent $\boldsymbol{w}^c_{0,3:5}$ and $\boldsymbol{w}^l_{0,3:5}$ as a linear span of basis vectors $\boldsymbol{\ell}_1$ and $\boldsymbol{\ell}_2$ as follows:

$$\boldsymbol{w}^c_{0,3:5} = e_{c_1}\boldsymbol{\ell}_1 + e_{c_2}\boldsymbol{\ell}_2, \quad \boldsymbol{w}^l_{0,3:5} = e_{l_1}\boldsymbol{\ell}_1 + e_{l_2}\boldsymbol{\ell}_2. \tag{4.26}$$

Importantly, the orthonormal transform matrix $\Psi_{T\times T}$ in (4.20) (i.e. an orthonormal basis in $\mathbb{R}^5$ in this example) is constructed by recursively updating its initial input $\Psi_0 = \mathbf{I}_{5\times 5}$ through local orthonormal transforms. For example, if $(p,q,r)^{\text{th}}$ elements in

$\boldsymbol{s}$ are selected to be merged, then we extract the corresponding $(p, q, r)^{\text{th}}$ columns of $\Psi^\top$ and update them through the matrix multiplication with Λ used in that merge. Therefore, the first orthonormal transform performed in (4.25) updates the initial matrix $\Psi_0^\top$ by multiplying Λ to the corresponding $(3, 4, 5)^{th}$ columns of $\Psi_0^\top$ which returns the following,

$$\Psi^\top = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & \ell_{1,1} & \ell_{2,1} & a \\ 0 & 0 & \ell_{1,2} & \ell_{2,2} & b \\ 0 & 0 & \ell_{1,3} & \ell_{2,3} & c \end{pmatrix}. \tag{4.27}$$

The 5^{th} column of $\Psi^\top$ is now fixed (not going to be updated again) as it corresponds to the detail coefficient but other four columns corresponding to the smooth coefficients in $\boldsymbol{s}$ would be updated as the merging continues.

Second merge. Suppose that $(X_2, s^1_{3,4,5}, s^2_{3,4,5})$ are selected to be merged next under the "two together" rule. Then we need to find the following orthonormal transform matrix,

$$\Lambda^* = \begin{pmatrix} \ell^*_{1,1} & \ell^*_{1,2} & \ell^*_{1,3} \\ \ell^*_{2,1} & \ell^*_{2,2} & \ell^*_{2,3} \\ a^* & b^* & c^* \end{pmatrix} = \begin{pmatrix} \boldsymbol{\ell}_1^{*\top} \\ \boldsymbol{\ell}_2^{*\top} \\ \boldsymbol{h}^{*\top} \end{pmatrix}, \tag{4.28}$$

where its elements would be different from those in (4.24). The detail filter $\boldsymbol{h}^{*\top} = (a^*, b^*, c^*)$ is constructed from the corresponding weight vectors, $\boldsymbol{w}^c_{2:4} = (1, e_{c_1}, e_{c_2})^\top$ and $\boldsymbol{w}^l_{2:4} = (2, e_{l_1}, e_{l_2})^\top$, by satisfying the conditions (1) $\boldsymbol{h}^{*\top}\boldsymbol{w}^c_{2:4} = 0$, (2) $\boldsymbol{h}^{*\top}\boldsymbol{w}^l_{2:4} = 0$ and (3) $\boldsymbol{h}^{*\top}\boldsymbol{h}^* = 1$. The detail filter is a weight vector designed for indicating the strength of linearity in (X_2, X_3, X_4, X_5) as (e_{c_1}, e_{c_2}) and (e_{l_1}, e_{l_2}) already contain the information of three raw observations (X_3, X_4, X_5). Then, two low filters, $\boldsymbol{\ell}^*_1$ and $\boldsymbol{\ell}^*_2$, are

obtained by satisfying the conditions, $\boldsymbol{\ell}_1^{*\top}\boldsymbol{h}^* = 0$, $\boldsymbol{\ell}_2^{*\top}\boldsymbol{h}^* = 0$, $\boldsymbol{\ell}_1^{*\top}\boldsymbol{\ell}_2^* = 0$ and ${\Lambda^*}^\top\Lambda^* = \mathbf{I}$. Now the data sequence and the weight vectors are updated as follows,

$$\begin{aligned} \boldsymbol{s} = (X_1, X_2, s^1_{3,5}, s^2_{3,5}, d_{3,4,5}) \quad &\rightarrow \quad \boldsymbol{s} = (X_1, s^1_{2,5}, s^2_{2,5}, d_{2,2,5}, d_{3,4,5}), \\ \boldsymbol{w}_c = (1, 1, e_{c_1}, e_{c_2}, 0)^\top \quad &\rightarrow \quad \boldsymbol{w}_c = (1, e^*_{c_1}, e^*_{c_2}, 0, 0)^\top, \\ \boldsymbol{w}_l = (1, 2, e_{l_1}, e_{l_2}, 0)^\top \quad &\rightarrow \quad \boldsymbol{w}_l = (1, e^*_{l_1}, e^*_{l_2}, 0, 0)^\top, \end{aligned} \tag{4.29}$$

and $\Psi^\top$ is also updated into

$$\Psi^\top = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & \ell^*_{1,1} & \ell^*_{2,1} & a^* & 0 \\ 0 & \begin{pmatrix} \\ \ell^*_{1,2}\boldsymbol{\ell}_1 + \ell^*_{1,3}\boldsymbol{\ell}_2 \\ \\ \end{pmatrix} & \begin{pmatrix} \\ \ell^*_{2,2}\boldsymbol{\ell}_1 + \ell^*_{2,3}\boldsymbol{\ell}_2 \\ \\ \end{pmatrix} & \begin{pmatrix} \\ b^*\boldsymbol{\ell}_1 + c^*\boldsymbol{\ell}_2 \\ \\ \end{pmatrix} & a \\ 0 & & & & b \\ 0 & & & & c \end{pmatrix}. \tag{4.30}$$

At this scale, the 4^{th} column of $\Psi^\top$ is fixed. This corresponds to the Type 2 basis vector in (4.41) whose non-zero subregion is composed of a single point (a^*) and a linear trend ($b^*\boldsymbol{\ell}_1 + c^*\boldsymbol{\ell}_2$).

Importantly, the orthonormal transform at this scale is performed in a way of returning an orthonormal basis of the expanded subspace e.g. 2^{nd} and 3^{rd} columns of (4.30) (which are referred to as $\boldsymbol{\ell}_1^{**}$ and $\boldsymbol{\ell}_2^{**}$ in (4.31)) are obtained as an arbitrary orthonormal basis of the subspace $\{(w, x, y, z) \mid w - x = x - y = y - z\}$ of $\mathbb{R}^4$. This is due to the semi-orthogonality of the transformation matrix $\boldsymbol{\Pi}$ in (4.31) which extends the dimension from $\mathbb{R}^3$ to $\mathbb{R}^4$ but preserves the fact that $(\boldsymbol{\ell}_1^*, \boldsymbol{\ell}_2^*)$ and $(\boldsymbol{\ell}_1^{**}, \boldsymbol{\ell}_2^{**})$ form an arbitrary orthonormal basis of the corresponding subspaces. This guarantees the

properties, ${\boldsymbol{\ell}_1^{**}}^\top \boldsymbol{\ell}_2^{**} = 0$ and ${\boldsymbol{\ell}_1^{**}}^\top \boldsymbol{\ell}_1^{**} = {\boldsymbol{\ell}_2^{**}}^\top \boldsymbol{\ell}_2^{**} = 1$, where

$$\boldsymbol{\ell}_1^{**} = \begin{pmatrix} \ell_{1,1}^* \\ \begin{pmatrix} \\ \ell_{1,2}^*\boldsymbol{\ell}_1 + \ell_{1,3}^*\boldsymbol{\ell}_2 \\ \\ \end{pmatrix} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \begin{pmatrix} \\ \boldsymbol{\ell}_1 \\ \\ \end{pmatrix} & \begin{pmatrix} \\ \boldsymbol{\ell}_2 \\ \\ \end{pmatrix} \\ 0 & & \\ 0 & & \end{pmatrix} \begin{pmatrix} \ell_{1,1}^* \\ \ell_{1,2}^* \\ \ell_{1,3}^* \end{pmatrix} = \boldsymbol{\Pi} \begin{pmatrix} \ell_{1,1}^* \\ \ell_{1,2}^* \\ \ell_{1,3}^* \end{pmatrix},$$
$$\boldsymbol{\ell}_2^{**} = \begin{pmatrix} \ell_{2,1}^* \\ \begin{pmatrix} \\ \ell_{2,2}^*\boldsymbol{\ell}_1 + \ell_{2,3}^*\boldsymbol{\ell}_2 \\ \\ \end{pmatrix} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \begin{pmatrix} \\ \boldsymbol{\ell}_1 \\ \\ \end{pmatrix} & \begin{pmatrix} \\ \boldsymbol{\ell}_2 \\ \\ \end{pmatrix} \\ 0 & & \\ 0 & & \end{pmatrix} \begin{pmatrix} \ell_{2,1}^* \\ \ell_{2,2}^* \\ \ell_{2,3}^* \end{pmatrix} = \boldsymbol{\Pi} \begin{pmatrix} \ell_{2,1}^* \\ \ell_{2,2}^* \\ \ell_{2,3}^* \end{pmatrix}, \tag{4.31}$$

and $\boldsymbol{\Pi}$ is obtained from the 2nd to 4th columns of (4.27) and the selected rows correspond to the indices of smooth coefficients associated in the orthonormal transformation in (4.28).

As is in (4.26), now the extended subregions of the original weight vectors, $\boldsymbol{w}_{0,2:5}^c$ and $\boldsymbol{w}_{0,2:5}^l$, can also be presented as a linear combination of $\boldsymbol{\ell}_1^{**}$ and $\boldsymbol{\ell}_2^{**}$ as follows:

$$\boldsymbol{w}_{0,2:5}^c = e_{c_1}^* \boldsymbol{\ell}_1^{**} + e_{c_2}^* \boldsymbol{\ell}_2^{**}, \quad \boldsymbol{w}_{0,2:5}^l = e_{l_1}^* \boldsymbol{\ell}_1^{**} + e_{l_2}^* \boldsymbol{\ell}_2^{**}, \tag{4.32}$$

where $\boldsymbol{\ell}_1^{**}$ and $\boldsymbol{\ell}_2^{**}$ form an orthonormal basis of the subspace $\{(w, x, y, z) \mid w - x = x - y = y - z\}$ of $\mathbb{R}^4$. This can be simply shown by 1) expressing the weight vectors as a linear combination of two low filters,

$$\begin{aligned} \boldsymbol{w}_{2:4}^c &= (1, e_{c_1}, e_{c_2})^\top = e_{c_1}^* \boldsymbol{\ell}_1^* + e_{c_2}^* \boldsymbol{\ell}_2^*, \\ \boldsymbol{w}_{2:4}^l &= (2, e_{l_1}, e_{l_2})^\top = e_{l_1}^* \boldsymbol{\ell}_1^* + e_{l_2}^* \boldsymbol{\ell}_2^*, \end{aligned} \tag{4.33}$$

and 2) performing the matrix multiplication with $\mathbf{\Pi}$ in (4.31) to both sides of (4.33),

$$\begin{aligned}
&\text{LHS} : \mathbf{\Pi}\boldsymbol{w}^c_{2:4} = (1, e_{c_1}\boldsymbol{\ell}_1 + e_{c_2}\boldsymbol{\ell}_2)^\top = (1,1,1,1)^\top = \boldsymbol{w}^c_{0,2:5}, \quad \text{RHS} : e^*_{c_1}\boldsymbol{\ell}^{**}_1 + e^*_{c_2}\boldsymbol{\ell}^{**}_2, \\
&\text{LHS} : \mathbf{\Pi}\boldsymbol{w}^l_{2:4} = (2, e_{l_1}\boldsymbol{\ell}_1 + e_{l_2}\boldsymbol{\ell}_2)^\top = (2,3,4,5)^\top = \boldsymbol{w}^l_{0,2:5}, \quad \text{RHS} : e^*_{l_1}\boldsymbol{\ell}^{**}_1 + e^*_{l_2}\boldsymbol{\ell}^{**}_2.
\end{aligned} \tag{4.34}$$

Last merge. In the same manner, after the last orthonormal transform is applied to $(X_1, s^1_{2,5}, s^2_{2,5})$, we end up with the finalised $\Psi^\top$ in which an orthonormal basis of the subspace $\{(v,w,x,y,z) \mid v - w = w - x = x - y = y - z\}$ of $\mathbb{R}^5$ is shown in its first and second columns where these two columns correspond to two basis vectors, $\psi^{(0,1)}$ and $\psi^{(0,2)}$, in (4.20). Regardless of the length of data (T), the first two columns of the finalised $\Psi^\top$ build two smooth coefficients $(s^1_{1,T}, s^2_{1,T})$ and always keep a linear trend with length T, while the shape of other columns of $\Psi^\top$ corresponding to the detail coefficients depends on the type of merge and follows one of the forms in (4.41).

As shown above, the non-uniqueness of the low filters has no effect on preserving the linearity of the subregions that are already merged. In simulation studies, we empirically found that the choice of low filters has no qualitative effect on the results as long as they are chosen by satisfying the orthonormality condition of the transform, thus we used a fixed type of function for choosing a set of low filters rather than choosing an arbitrary set of low filters that satisfies the orthonormal condition every run which also saves the computational costs.

4.3 Theoretical results

We study the l_2 consistency of $\tilde{\boldsymbol{f}}$ and $\tilde{\tilde{\boldsymbol{f}}}$, and the change-point detection consistency of $\hat{\boldsymbol{f}}$, where the estimators are defined in Section 4.2. The l_2 risk of an estimator $\tilde{\boldsymbol{f}}$ is defined as $\left\|\tilde{\boldsymbol{f}} - \boldsymbol{f}\right\|^2_T = T^{-1}\sum_{i=1}^T(\tilde{f}_i - f_i)^2$, where $\boldsymbol{f}$ is the underlying signal as in (4.1).

We note the true change-points $\{\eta_i, i = 1, \ldots, N\}$ are such that,

$$\begin{aligned} f_t &= \theta_{\ell,1} + \theta_{\ell,2}\, t \quad \text{for} \quad t \in [\eta_{\ell-1} + 1, \eta_\ell], \ \ell = 1, \ldots, N+1 \\ &\text{where} \quad f_{\eta_\ell} + \theta_{\ell,2} \neq f_{\eta_\ell+1} \quad \text{for} \quad \ell = 1, \ldots, N. \end{aligned} \tag{4.35}$$

This definition permits both continuous and discontinuous changes and if f_{η_i} is a point anomaly, there exist two consecutive change-points at $\eta_i - 1$ and η_i where $\eta_{i-1} = \eta_i - 1$. The consistency of the estimated number and locations of the change-points is established under the following conditions.

Assumption 4.1 $\sigma^2 = Var(\varepsilon_t) = 1$ *in model* (4.1).

Assumption 4.2 *Let the threshold take the form of* $\lambda = C_1\{2\log(T)\}^{1/2}$ *with a constant* C_1 *large enough.*

In Assumption 4.1, σ is assumed to be known for simplicity as it is a nuisance parameter. If it is unknown, we can plug in the Median Absolute Deviation estimator (Hampel, 1974) that will be formulated later in Section 4.4.1 and this does not affect the validity of our theory. However, here we assume $\sigma = 1$ for notational convenience which is standard in the literature.

Assumption 4.2 is established on Assumption 4.1 where the generalised form of the threshold is $\lambda = C_1\sigma\{2\log(T)\}^{1/2}$. The optimal value of the constant C_1 for the practical application of TrendSegment procedure will be specified in Section 4.4.1. Regarding the degree of dependence on Gaussianity, the normality assumption plays an important role in Lemma 4.1 in Section 4.6 in that the tail bound for standard normal distribution is associated with the size of the threshold λ in Assumption 4.2. It is reasonable to consider the case when the i.i.d Gaussian assumption on ε_t is extended to dependent or heavy-tailed noise, which we do not pursue in this section but an extension to dependent noise will be explored in Chapter 5.

We firstly investigate the l_2 behaviour of $\tilde{\boldsymbol{f}}$. The proofs of Theorems 4.1-4.3 can be found in Section 4.6.

Theorem 4.1 *X_t follows model (4.1) and $\tilde{\boldsymbol{f}}$ is the estimator in (4.17). Then under Assumptions 4.1-4.2, we have*

$$\|\tilde{\boldsymbol{f}} - \boldsymbol{f}\|_T^2 \leq C_1^2 \frac{1}{T} \log(T) \left\{ 4 + 8N \left\lceil \log(T)/\log\left((1-\rho)^{-1}\right) + \log(2)/\log(1-\rho) \right\rceil \right\}, \tag{4.36}$$

with probability approaching 1 as $T \to \infty$ and the piecewise-linear estimator $\tilde{\boldsymbol{f}}$ contains $\tilde{N} \leq CN\log(T)$ change-points where C is a constant and $\rho \in (0,1)$ is a constant which controls the greediness level of the TGUW transformation.

Thus, $\tilde{\boldsymbol{f}}$ is l_2 consistent if $N = O(1)$. The crucial mechanism of l_2 consistency is the "tail-greediness" which allows up to $K(j) \geq 1$ smooth coefficients to be removed at each scale j. In other words, if we proceed in a greedy way, i.e. we only merge one triplet at each scale of the TGUW transformation, then l_2 consistency is generally unachievable as the largest number of scales J is not bounded above by $\left\lceil \log(T)/\log\left((1-\rho)^{-1}\right) + \log(2)/\log(1-\rho) \right\rceil$ in (4.36). Thus, in simulations and data application, we merge more than one triplet in a single pass over the data and the details can be found in Section 4.4.1.

We now move onto the estimator $\tilde{\tilde{\boldsymbol{f}}}$ obtained in the first stage of post-processing.

Theorem 4.2 *X_t follows model (4.1) and $\tilde{\tilde{\boldsymbol{f}}}$ is the estimator in (4.19). Then under Assumptions 4.1-4.2, we have $\left\|\tilde{\tilde{\boldsymbol{f}}} - \boldsymbol{f}\right\|_T^2 = O\left(NT^{-1}\log^2(T)\right)$ with probability approaching 1 as $T \to \infty$ and there exist at most two estimated change-points between each pair of true change-points (η_i, η_{i+1}) for $i = 0, \ldots, N$, where $\eta_0 = 0$ and $\eta_{N+1} = T$. Therefore $\tilde{\tilde{N}} \leq 2(N+1)$.*

We see that $\tilde{\tilde{\boldsymbol{f}}}$ is l_2 consistent, but inconsistent for the number of change-points. For the consistency of the estimated number and locations of the change-points, we consider the following conditions.

Assumption 4.3 *The number of true change-points, N, is finite.*

Assumption 4.4 *Let $\Delta_T = \min_{\ell=1,\ldots,N}\left\{\left(\underline{f}_T^{\ell}\right)^{2/3}\cdot\delta_T^{\ell}\right\}$ where $\underline{f}_T^{\ell} = \min\left(\left|f_{\eta_{\ell+1}} - 2f_{\eta_\ell} + f_{\eta_{\ell-1}}\right|, \left|f_{\eta_{\ell+2}} - 2f_{\eta_{\ell+1}} + f_{\eta_\ell}\right|\right)$ and $\delta_T^{\ell} = \min\left(\left|\eta_\ell - \eta_{\ell-1}\right|, \left|\eta_{\ell+1} - \eta_\ell\right|\right)$. Assume that $T^{1/3}R_T^{1/3} = o\left(\Delta_T\right)$ where $\left\|\tilde{\tilde{\boldsymbol{f}}} - \boldsymbol{f}\right\|_T^2 = O_p\left(R_T\right)$ is as in Theorem 4.2.*

Assumption 4.3 controls the number of true change-points, which is often used (see e.g. Dalalyan et al. (2017), Fryzlewicz (2018b)) and sometimes called the "strong sparsity" case, in contrast to the "weak sparsity" case, in which the total variation of the true signal is controlled. Assumption 4.4 quantifies the difficulty of detecting a change-point in terms of distance from its neighbouring change-points and size of the change in linear trend.

Now we investigate the final estimators, $\hat{\boldsymbol{f}}$ and $\hat{N}$.

Theorem 4.3 *X_t follows model (4.1) and $(\hat{\boldsymbol{f}}, \hat{N})$ are the estimators obtained in Section 4.2.5. Then under Assumptions 4.1-4.4, we have*

$$\mathbb{P}\left(\hat{N} = N, \quad \max_{\ell=1,\ldots,N}\left\{|\hat{\eta}_\ell - \eta_\ell|\cdot\left(\underline{f}_T^{\ell}\right)^{2/3}\right\} \leq CT^{1/3}R_T^{1/3}\right) \;\to\; 1, \tag{4.37}$$

as $T\to\infty$ where C is a constant.

Our theory indicates that in the case in which $\min_\ell \underline{f}_T^{\ell}$ is bounded away from zero, the consistent estimation of the number and locations of change-point is achieved by assuming $T^{1/3}R_T^{1/3} = o(\delta_T)$ where $\delta_T = \min_{\ell=1,\ldots,N+1}|\eta_\ell - \eta_{\ell-1}|$. In addition, when point anomalies exist in the set of true change-points, a point anomaly η_k and its neighbouring change-point $\eta_{k-1} = \eta_k - 1$ can be detected exactly at their true locations only if the corresponding $\underline{f}_T^{\ell}$s satisfy the condition $\min\left(\underline{f}_T^{k}, \underline{f}_T^{k-1}\right) \gtrsim \log(T)$.

4.4 Simulation study

4.4.1 Parameter choice

Choice of the "tail-greediness" parameter. As introduced in Section 4.2.2, $\rho \in (0, 1)$ is a constant which controls the greediness level of the TGUW transformation in the sense that it decides how many merges are performed in a single pass over the data. A large ρ can reduce the computational cost but it makes the procedure less adaptive, whereas a small ρ gives the opposite effect. Based on our empirical experience, the best performance is achieved in the range $\rho \in (0, 0.05]$ and we use $\rho = 0.04$ as a default in the simulation study and data analyses.

Choice of threshold λ. Motivated by Theorem 4.1, we use the threshold of the form $\lambda = C\sigma(2\log T)^{1/2}$ and estimate σ using the Median Absolute Deviation (MAD) estimator (Hampel, 1974) defined as $\hat{\sigma} = \text{Median}(|X_1 - 2X_2 + X_3|, \ldots, |X_{T-2} - 2X_{T-1} + X_T|)/(\Phi^{-1}(3/4)\sqrt{6})$ where Φ^{-1} is the quantile function of the Gaussian distribution. We use $C = 1.3$ as a default as it empirically led to the best performance over the range $C \in [1, 1.4]$. As we can find a threshold which corresponds to a specific candidate model produced by TrendSegment, in practice, the user can set the threshold in a way of finding the best model (i.e. best for their goal from their point of view), from all possible candidate models. We illustrate how the choice of the threshold affects the estimated change-points in Section 4.5.1 with Iceland temperature data.

Choice of the parameter for balancedness. As our wavelet basis is unbalanced, we define the parameter $\mathcal{B}$ balancing the estimated change-points which makes $\tilde{\eta}_i$ to be survived from thresholding only when the following condition is satisfied,

$$\mathcal{B} < \frac{\tilde{\eta}_{i+1} - \tilde{\eta}_i}{\tilde{\eta}_{i+1} - \tilde{\eta}_{i-1}} < 1 - \mathcal{B}, \quad \text{where} \quad i = 1, \ldots, \tilde{N} \quad \text{and} \quad \mathcal{B} \in [0, 1/2), \tag{4.38}$$

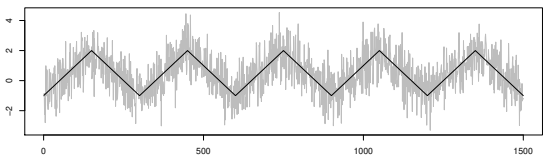
with the convention $\tilde{\eta}_0 = 0$ and $\tilde{\eta}_{\tilde{N}+1} = T$. In the remainder of the chapter, we use $\mathcal{B} = 0$ as a default as it allows the TrendSegment procedure to detect both point anomalies and change-points in linear trend at once. Someone who is interested only in detecting (relatively long) linear trends without point anomalies can control this balancing parameter in the R package `trendsegmentR` as some extent of balancing can improve the accuracy of estimating the number of change-points.

4.4.2 Simulation settings

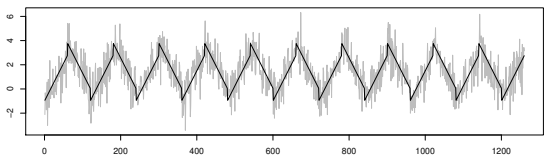
We consider i.i.d. Gaussian noise and simulate data from model (4.1) using 8 signals, (M1) wave1, (M2) wave2, (M3) mix1, (M4) mix2, (M5) mix3, (M6) lin.sgmts, (M7) teeth and (M8) lin, shown in Figure 4.4. (M1) is continuous at change-points, while (M2) has discontinuities. (M3) has a mix of continuous and discontinuous change-points and contains both constant and linear segments, whereas (M4) is of the same type but also contains two point anomalies. In addition, (M5) has two particularly short segments. (M6) contains isolated spike-type short segments. (M7) is piecewise-constant, and (M8) is a linear signal without change-points. We note that the simulation results under dependent or heavy-tailed errors are also presented and the signals and R code for all simulations can be downloaded from our GitHub repository (Maeng, 2019a).

4.4.3 Competing methods and estimators

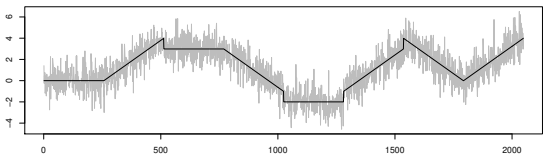
We perform the TrendSegment procedure based on the parameter choice in Section 4.4.1 and compare the performance with that of the following competitors: Narrowest-Over-Threshold detection (**NOT**, Baranowski et al. (2019)) implemented in the R package `not` from CRAN, Isolate-Detect (**ID**, Anastasiou and Fryzlewicz (2019)) available in the R package `IDetect`, trend filtering (**TF**, Kim et al. (2009)) available from https://github.com/glmgen/genlasso, Continuous-piecewise-linear Pruned Optimal



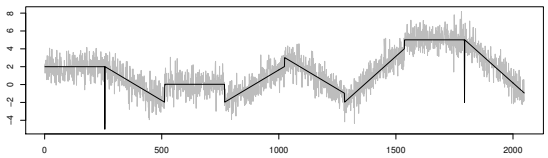
(a) (M1) wave1



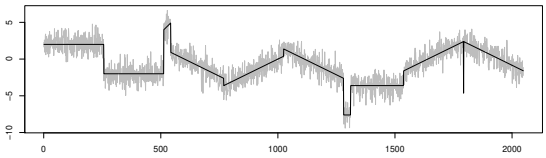
(b) (M2) wave2



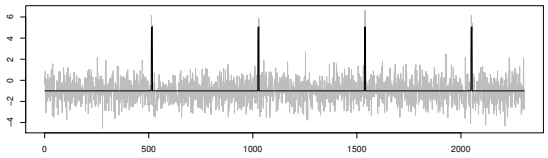
(c) (M3) mix1



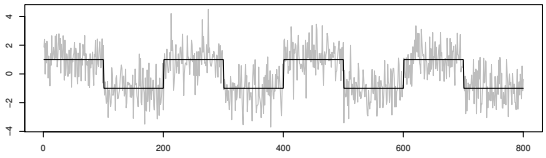
(d) (M4) mix2



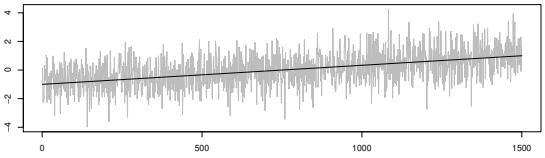
(e) (M5) mix3



(f) (M6) lin.sgmts



(g) (M7) teeth



(h) (M8) lin

Fig. 4.4 Examples of data with its underlying signal studied in Section 4.4. (a)-(h) data series X_t (light grey) and true signal f_t (black).

Partitioning (**CPOP**, Maidstone et al. (2017a)) available from https://www.maths.lancs.ac.uk/~fearnhea/Publications.html and a bottom-up algorithm based on the residual sum of squares (RSS) from a linear fit (**BUP**, Keogh et al. (2004)). The TrendSegment methodology is implemented in the R package `trendsegmentR`.

As BUP requires a pre-specified number of change-points or a well-chosen stopping criterion which can vary depending on the data, we include it in the simulation study with the stopping criterion optimised for the best performance using the knowledge of the truth but do not include it in data applications. We are aware that the methods of Spiriti et al. (2013) and Bai and Perron (2003) implemented in the R packages

`freeknotsplines` and `strucchange` can be added in a list of competing methods however, these are excluded as we have found them to be particularly slow. For instance, the minimum segment size in `strucchange` can be adjusted to be small as long as it is greater than or equal to 3 for detecting linear trend changes. This cannot capture point anomalies but is suitable for detecting very short segments (e.g in (M6) lin.sgmts). However, this setting is accompanied by extremely heavy computation: with this minimum segment size in place, a single signal simulated from (M6) took us over three hours to process on a standard PC.

Out of the competing methods tested, ID, TF and CPOP are in principle able to classify two consecutive time point as change-points, and therefore they are able to detect point anomalies. NOT and BUP are not designed to detect point anomalies as their minimum distance between two consecutive change-points is restricted to be at least two. For NOT, we use the contrast function for not necessarily continuous piecewise-linear signals. Regarding the tuning parameters for the competing methods, we follow the recommendation of each respective paper or the corresponding R package.

4.4.4 Results

The summary of the results for all models and methods can be found in Tables 4.2 and 4.3. We run 100 simulations and as a measure of the accuracy of estimators, we use Monte-Carlo estimates of the Mean Squared Error of the estimated signal defined as MSE=$\mathbb{E}\{(1/T)\sum_{t=1}^{T}(f_t - \hat{f}_t)^2\}$. The empirical distribution of $\hat{N} - N$ is also reported where $\hat{N}$ is the estimated number of change-points and N is the true one. In addition to this, for comparing the accuracy of the locations of the estimated change-points $\hat{\eta}_i$, we show estimates of the scaled Hausdorff distance given by

$$d_H = \frac{1}{T}\mathbb{E}\max\left\{\max_i \min_j \left|\eta_i - \hat{\eta}_j\right|, \quad \max_j \min_i \left|\hat{\eta}_j - \eta_i\right|\right\} \tag{4.39}$$

where $i = 0, \ldots, N+1$ and $j = 0, \ldots, \hat{N}+1$ with the convention $\eta_0 = \hat{\eta}_0 = 0, \eta_{N+1} = \hat{\eta}_{N+1} = T$ and $\hat{\eta}$ and η denote estimated and true locations of the change-points. The smaller the Hausdorff distance, the better the estimation of the change-point locations. For each method, the average computation time in seconds is shown.

The results for (M1) and (M2) are similar. TrendSegment shows comparable performance to NOT, ID and CPOP in terms of the estimation of the number of change-points while it is less attractive in terms of the estimated locations of change-points. TF tends to overestimate the number of change-points throughout all models. When the signal is a mix of constant and linear trends as in (M3), TrendSegment, NOT and ID still perform well in terms of the estimation of the number of change-points while CPOP tends to overestimate. We see that TrendSegment has a particular advantage over the other methods especially in (M4) and (M5), when point anomalies exist or in the case of frequent change-points. TrendSegment shows its relative robustness in estimating the number and the location of change-points while ID and CPOP significantly underperform and NOT completely ignores the point anomalies as expected. (M6) is another example where only TrendSegment shows a good performance. For the estimation of the piecewise-constant signal (M7), no methods show good performances and NOT, ID and TrendSegment tend to underestimate the number of change-points while CPOP and TF overestimate. In the case of the no-change-point signal (M8), all methods estimate well except TF. In summary, TrendSegment is never among the worst methods, is almost always among the best ones, and is particularly attractive for signals with point anomalies or short segments. With respect to computation time, NOT and ID are very fast in all cases, TrendSegment is slower than these two but is faster than TF, CPOP and BUP, especially when the length of the time series is larger than 2000.

Table 4.2 Distribution of $\hat{N} - N$ for models (M1)-(M4) and all methods listed in Sections 4.4.1 and 4.4.3 with i.i.d. Gaussian noise over 100 simulation runs. Also the average MSE (Mean Squared Error) of the estimated signal $\hat{f}_t$ defined in Section 4.4.4, the average Hausdorff distance d_H given by (4.39) and the average computational time in seconds using an Intel Core i5 2.9 GHz CPU with 8 GB of RAM, all over 100 simulations. Bold: methods within 10% of the highest empirical frequency of $\hat{N} - N = 0$ or within 10% of the lowest empirical average $d_H(\times 10^2)$. Note that TrendSegment is shortened to TS.

		$\hat{N} - N$									
Model	Method	≤-3	-2	-1	0	1	2	≥3	MSE	$d_H(\times 10^2)$	time
(M1)	TS	0	0	0	**99**	1	0	0	0.044	2.79	1.12
	NOT	0	0	0	**99**	1	0	0	0.034	2.09	0.29
	ID	0	0	0	**99**	1	0	0	0.029	1.45	0.22
	TF	0	0	0	0	0	0	100	0.016	4.29	36.30
	CPOP	0	0	0	**99**	1	0	0	0.014	**0.78**	8.55
	BUP	0	1	18	81	0	0	0	0.069	3.88	2.62
(M2)	TS	0	0	2	**98**	0	0	0	0.109	1.90	1.06
	NOT	0	0	2	**98**	0	0	0	0.092	1.56	0.35
	ID	0	0	0	**94**	6	0	0	0.089	**1.44**	0.23
	TF	0	0	0	0	0	0	100	0.065	2.31	31.34
	CPOP	0	0	0	**93**	7	0	0	0.065	**1.15**	2.09
	BUP	100	0	0	0	0	0	0	0.752	4.69	2.21
(M3)	TS	0	0	1	**97**	2	0	0	0.032	3.23	1.47
	NOT	0	0	0	**100**	0	0	0	0.020	**2.35**	0.36
	ID	0	0	1	**94**	5	0	0	0.047	**2.37**	0.33
	TF	0	0	0	0	0	0	100	0.023	5.87	45.31
	CPOP	0	0	0	61	32	6	1	0.024	**2.34**	21.11
	BUP	0	0	0	3	18	47	32	0.041	5.41	3.50
(M4)	TS	0	0	5	**76**	18	1	0	0.030	**1.81**	1.48
	NOT	0	100	0	0	0	0	0	0.066	2.10	0.33
	ID	0	11	52	35	2	0	0	0.163	**1.83**	0.30
	TF	0	0	0	0	0	0	100	0.080	6.10	44.78
	CPOP	0	0	2	22	45	27	4	0.025	**1.60**	7.79
	BUP	0	0	8	31	45	13	3	0.092	5.30	3.62

Table 4.3 Distribution of $\hat{N} - N$ for models (M5)-(M8) and all methods listed in Sections 4.4.1 and 4.4.3 with i.i.d. Gaussian noise over 100 simulation runs. Also the average MSE (Mean Squared Error) of the estimated signal $\hat{f}_t$ defined in Section 4.4.4, the average Hausdorff distance d_H given by (4.39) and the average computational time in seconds using an Intel Core i5 2.9 GHz CPU with 8 GB of RAM, all over 100 simulations. Bold: methods within 10% of the highest empirical frequency of $\hat{N} - N = 0$ or within 10% of the lowest empirical average $d_H(\times 10^2)$. Note that TrendSegment is shortened to TS.

		$\hat{N} - N$									
Model	Method	≤-3	-2	-1	0	1	2	≥3	MSE	$d_H(\times 10^2)$	time
(M5)	TS	0	0	1	**71**	24	4	0	0.031	**1.42**	1.49
	NOT	0	0	99	1	0	0	0	0.040	**1.20**	0.29
	ID	0	0	1	2	14	32	51	0.277	8.28	0.30
	TF	0	0	0	0	0	0	100	0.116	6.17	43.13
	CPOP	0	0	0	11	22	39	28	0.023	**1.41**	5.12
	BUP	0	0	10	45	37	7	1	0.090	4.78	3.64
(M6)	TS	0	0	0	**96**	4	0	0	0.013	**0.05**	1.65
	NOT	63	22	4	2	3	0	6	0.240	15.51	0.28
	ID	3	16	0	9	44	1	27	0.151	16.37	0.37
	TF	0	0	0	0	0	0	100	0.134	10.98	48.19
	CPOP	0	0	0	20	41	24	15	0.034	**0.13**	5.11
	BUP	0	0	0	0	0	0	100	0.135	10.17	4.00
(M7)	TS	0	5	21	40	28	6	0	0.119	7.02	0.65
	NOT	1	1	8	**56**	31	3	0	0.065	**2.62**	0.25
	ID	3	0	16	14	26	13	28	0.320	10.87	0.12
	TF	0	0	0	0	0	0	100	0.097	6.11	23.19
	CPOP	0	0	1	1	3	17	78	0.055	**3.37**	1.19
	BUP	70	25	5	0	0	0	0	0.277	11.89	1.58
(M8)	TS	0	0	0	**100**	0	0	0	0.001	**0.00**	1.01
	NOT	0	0	0	**100**	0	0	0	0.001	**0.00**	0.17
	ID	0	0	0	**100**	0	0	0	0.001	**0.00**	0.59
	TF	0	0	0	78	5	2	15	0.002	9.08	35.79
	CPOP	0	0	0	**100**	0	0	0	0.001	**0.00**	12.96
	BUP	0	0	0	0	0	0	100	0.011	46.34	2.63

Table 4.4 Distribution of $\hat{N} - N$ for models (M1)-(M4) and all methods listed in Sections 4.4.1 and 4.4.3 with the noise term ε_t being $AR(1)$ process of $\phi = 0.3$ over 100 simulation runs. Also the average MSE (Mean Squared Error) of the estimated signal $\hat{f}_t$, the average Hausdorff distance d_H and the average computational time in seconds using an Intel Core i5 2.9 GHz CPU with 8 GB of RAM, all over 100 simulations. Bold: methods within 10% of the highest empirical frequency of $\hat{N} - N = 0$ or within 10% of the lowest empirical average $d_H(\times 10^2)$. Note that TrendSegment is shortened to TS.

		$\hat{N} - N$									
Model	Method	≤-3	-2	-1	0	1	2	≥3	MSE	$d_H(\times 10^2)$	time
(M1)	TS	0	0	4	**93**	3	0	0	0.081	3.58	1.27
	NOT	0	0	0	**91**	9	0	0	0.072	2.95	0.26
	ID	0	0	0	82	14	4	0	0.067	**2.65**	0.38
	TF	0	0	0	0	0	0	100	0.532	5.00	36.61
	CPOP	0	0	0	7	15	6	72	0.080	3.69	4.95
	BUP	0	0	8	86	6	0	0	0.077	3.66	2.75
(M2)	TS	1	6	23	69	1	0	0	0.195	2.44	1.12
	NOT	0	0	8	**83**	6	2	1	0.182	2.11	0.31
	ID	0	0	0	69	24	5	2	0.155	**1.75**	0.40
	TF	0	0	0	0	0	0	100	0.600	2.38	32.03
	CPOP	0	0	0	1	6	8	85	0.163	**1.98**	1.50
	BUP	100	0	0	0	0	0	0	0.717	4.63	2.39
(M3)	TS	0	1	5	**88**	6	0	0	0.052	4.16	1.56
	NOT	0	0	0	**89**	7	4	0	0.042	**3.40**	0.31
	ID	0	0	3	77	16	3	1	0.064	**3.12**	0.50
	TF	0	0	0	0	0	0	100	0.259	6.24	44.94
	CPOP	0	0	0	1	4	10	85	0.068	4.67	9.57
	BUP	0	0	0	0	3	18	79	0.056	5.57	3.56
(M4)	TS	0	6	23	**53**	18	0	0	0.058	**2.41**	1.53
	NOT	0	93	6	1	0	0	0	0.086	2.91	0.31
	ID	2	6	30	49	10	2	1	0.165	2.99	0.48
	TF	0	0	0	0	0	0	100	0.218	6.22	45.99
	CPOP	0	0	0	1	3	9	87	0.066	4.02	5.40
	BUP	0	0	0	11	35	37	17	0.109	5.64	3.77

Table 4.5 Distribution of $\hat{N} - N$ for models (M5)-(M8) and all methods listed in Sections 4.4.1 and 4.4.3 with the noise term ε_t being $AR(1)$ process of $\phi = 0.3$ over 100 simulation runs. Also the average MSE (Mean Squared Error) of the estimated signal $\hat{f}_t$, the average Hausdorff distance d_H and the average computational time in seconds using an Intel Core i5 2.9 GHz CPU with 8 GB of RAM, all over 100 simulations. Bold: methods within 10% of the highest empirical frequency of $\hat{N} - N = 0$ or within 10% of the lowest empirical average $d_H(\times 10^2)$. Note that TrendSegment is shortened to TS.

		$\hat{N} - N$									
Model	Method	≤-3	-2	-1	0	1	2	≥3	MSE	$d_H(\times 10^2)$	time
(M5)	TS	0	0	19	**54**	21	6	0	0.055	**1.87**	1.54
	NOT	0	0	91	6	3	0	0	0.060	**1.94**	0.28
	ID	0	0	9	23	23	18	27	0.402	9.47	0.46
	TF	0	0	0	0	0	0	100	0.182	6.21	42.65
	CPOP	0	0	0	0	2	4	94	0.068	3.70	4.08
	BUP	0	0	0	15	37	32	16	0.112	5.25	3.53
(M6)	TS	0	0	0	**98**	2	0	0	0.018	**0.06**	1.70
	NOT	68	9	10	4	1	3	5	0.257	21.63	0.25
	ID	20	10	0	0	11	0	59	0.164	12.83	0.63
	TF	0	0	0	0	0	0	100	0.332	11.04	47.43
	CPOP	0	0	0	5	11	17	67	0.056	4.86	5.31
	BUP	0	0	0	0	0	0	100	0.170	10.18	3.95
(M7)	TS	11	38	31	15	3	2	0	0.217	11.52	0.68
	NOT	5	12	19	**24**	22	7	11	0.158	7.69	0.24
	ID	32	1	18	**26**	14	5	4	0.511	17.54	0.03
	TF	3	0	0	0	0	0	97	0.623	7.01	23.25
	CPOP	0	0	0	0	0	1	99	0.162	**5.27**	0.85
	BUP	54	43	3	0	0	0	0	0.283	11.92	1.55
(M8)	TS	0	0	0	**100**	0	0	0	0.003	**0.00**	1.09
	NOT	0	0	0	**93**	3	3	1	0.005	**2.02**	0.19
	ID	0	0	0	**100**	0	0	0	0.003	**0.00**	0.51
	TF	0	0	0	0	0	0	100	0.551	49.94	35.81
	CPOP	0	0	0	30	10	3	57	0.035	19.71	7.55
	BUP	0	0	0	0	0	0	100	0.025	46.73	2.72

Table 4.6 Distribution of $\hat{N} - N$ for models (M1)-(M4) and all methods listed in Sections 4.4.1 and 4.4.3 with the noise term ε_t being i.i.d. t_5 over 100 simulation runs. Also the average MSE (Mean Squared Error) of the estimated signal $\hat{f}_t$, the average Hausdorff distance d_H and the average computational time in seconds using an Intel Core i5 2.9 GHz CPU with 8 GB of RAM, all over 100 simulations. Bold: methods within 10% of the highest empirical frequency of $\hat{N} - N = 0$ or within 10% of the lowest empirical average $d_H(\times 10^2)$. Note that TrendSegment is shortened to TS.

		$\hat{N} - N$									
Model	Method	$\leq$-3	-2	-1	0	1	2	$\geq$3	MSE	$d_H(\times 10^2)$	time
(M1)	TS	0	0	3	55	32	5	5	0.083	3.41	1.09
	NOT	0	0	0	**98**	2	0	0	0.037	2.21	0.26
	ID	0	0	0	85	10	4	1	0.036	**1.87**	0.29
	TF	0	0	0	0	0	0	100	0.017	4.36	36.49
	CPOP	0	0	0	21	20	20	39	0.064	2.28	5.69
	BUP	0	4	13	78	5	0	0	0.071	3.84	2.62
(M2)	TS	0	3	11	70	10	5	1	0.164	2.18	1.03
	NOT	0	0	3	**85**	11	0	1	0.098	**1.69**	0.29
	ID	0	0	0	**77**	21	2	0	0.102	**1.36**	0.38
	TF	0	0	0	0	0	0	100	0.067	2.29	31.41
	CPOP	0	0	0	14	23	25	38	0.119	**1.54**	1.66
	BUP	100	0	0	0	0	0	0	0.752	4.69	2.18
(M3)	TS	0	1	11	41	25	10	12	0.073	4.90	1.44
	NOT	0	0	0	**96**	3	1	0	0.021	**2.54**	0.31
	ID	0	0	1	73	19	3	4	0.053	**2.72**	0.44
	TF	0	0	0	0	0	0	100	0.024	5.92	46.35
	CPOP	0	0	0	9	10	11	70	0.065	3.57	11.71
	BUP	0	0	0	1	21	40	38	0.043	5.44	3.52
(M4)	TS	0	3	14	34	23	16	10	0.075	3.10	1.46
	NOT	0	97	3	0	0	0	0	0.066	**2.45**	0.28
	ID	1	12	22	**48**	10	3	4	0.159	**2.42**	0.42
	TF	0	0	0	0	0	0	100	0.081	6.06	45.74
	CPOP	0	0	0	4	4	15	77	0.062	3.37	5.15
	BUP	0	2	7	28	47	12	4	0.095	5.30	3.56

Table 4.7 Distribution of $\hat{N} - N$ for models (M5)-(M8) and all methods listed in Sections 4.4.1 and 4.4.3 with the noise term ε_t being i.i.d. t_5 over 100 simulation runs. Also the average MSE (Mean Squared Error) of the estimated signal $\hat{f}_t$, the average Hausdorff distance d_H and the average computational time in seconds using an Intel Core i5 2.9 GHz CPU with 8 GB of RAM, all over 100 simulations. Bold: methods within 10% of the highest empirical frequency of $\hat{N} - N = 0$ or within 10% of the lowest empirical average $d_H(\times 10^2)$. Note that TrendSegment is shortened to TS.

		$\hat{N} - N$									
Model	Method	$\leq$-3	-2	-1	0	1	2	$\geq$3	MSE	$d_H(\times 10^2)$	time
(M5)	TS	0	0	9	**40**	19	21	11	0.069	2.72	1.49
	NOT	0	0	95	4	1	0	0	0.042	**1.29**	0.26
	ID	0	0	1	16	24	23	36	0.372	9.86	0.43
	TF	0	0	0	0	0	0	100	0.118	6.15	43.23
	CPOP	0	0	0	3	8	12	77	0.060	2.97	3.55
	BUP	0	0	10	**40**	43	6	1	0.083	4.76	3.51
(M6)	TS	0	0	0	**46**	2	39	13	0.035	**3.16**	1.66
	NOT	54	21	4	8	5	1	7	0.244	17.30	0.23
	ID	8	8	0	0	6	0	78	0.125	6.99	0.62
	TF	0	0	0	0	0	0	100	0.138	10.99	48.53
	CPOP	0	0	0	9	11	17	63	0.059	4.68	3.48
	BUP	0	0	0	0	0	0	100	0.145	10.27	3.92
(M7)	TS	14	28	32	14	8	4	0	0.204	11.21	0.65
	NOT	0	6	16	**30**	36	11	1	0.079	5.12	0.22
	ID	14	8	12	17	24	13	12	0.421	16.22	0.04
	TF	0	0	0	0	0	0	100	0.098	6.08	23.86
	CPOP	0	0	0	0	4	5	91	0.102	**3.01**	0.81
	BUP	69	28	3	0	0	0	0	0.266	12.12	1.47
(M8)	TS	0	0	0	49	0	43	8	0.030	14.86	1.01
	NOT	0	0	0	**100**	0	0	0	0.001	**0.00**	0.17
	ID	0	0	0	**99**	1	0	0	0.001	**0.00**	0.45
	TF	0	0	0	65	12	9	14	0.003	14.63	36.03
	CPOP	0	0	0	35	0	34	31	0.042	20.53	3.91
	BUP	0	0	0	0	0	0	100	0.014	46.80	2.62

In addition to the simulation results with i.i.d. Gaussian noise, we present the results with two different distributions of the noise, (a) ε_t follows a stationary Gaussian AR(1) process of $\phi = 0.3$, with zero-mean and unit-variance and (b) $\varepsilon_t \sim$ i.i.d. scaled t_5 distribution with unit-variance, where the summary of the results for all models and methods can be found in Tables 4.4-4.7. We use $C = 1.8$ as a default for the thresholding constant of TrendSegment. Among other competitors, only ID provides the option for heavy-tailed noise in their R package IDetect and other methods are set to their default settings. TrendSegment appears to be relatively useful under a heavy-tailed or dependent noise especially when the underlying signal contains point anomalies or short segments.

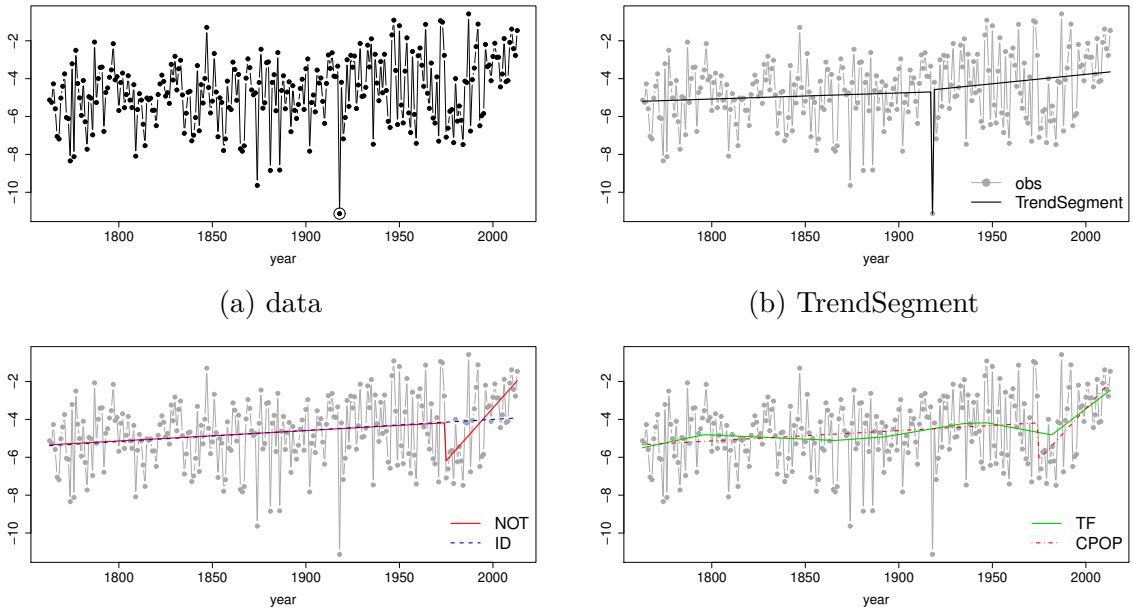
4.5 Data applications

4.5.1 Average January temperatures in Iceland

We analyse a land temperature dataset available from http://berkeleyearth.org, consisting of average temperatures in January recorded in Reykjavik recorded from 1763 to 2013. Figure 4.5a shows the data; the point corresponding to 1918 appears to be a point anomaly, where this aspect is commented earlier in Section 4.1.

The TrendSegment estimate of the piecewise-linear trend is shown in Figure 4.5b. It identifies 2 change-points, 1917 and 1918, where the temperature in 1918 is fitted as a single point as it is much lower than in other years. Figures 4.5c and 4.5d show that NOT and CPOP detect a change of slope in 1974, ID returns an increasing function with no change-points and TF reports 6 points with the most recent one in 1981, but none of them detects the point anomaly.

Out of the competing methods, all except NOT are in principle able to detect changes in linear trend and point anomalies at the same time. We examine whether any

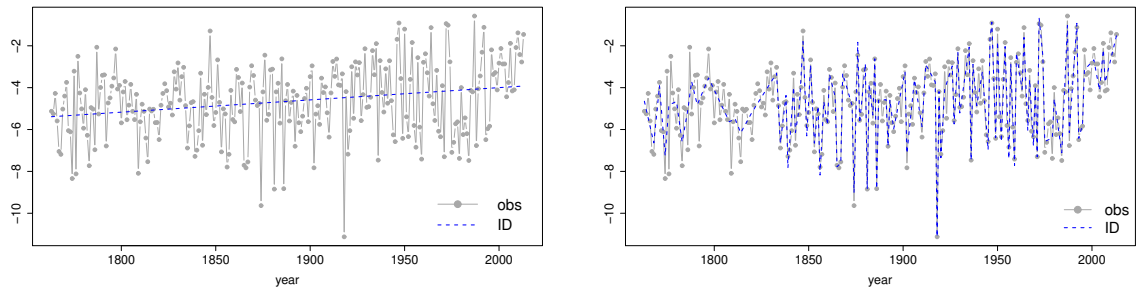


(a) data (b) TrendSegment

(c) NOT and ID (d) TF and CPOP

Fig. 4.5 Change-point analysis for January average temperature in Reykjavik from 1763 to 2013 in Section 4.5.1. (a) the data series, (b) the data series (grey dots) and estimated signal with change-points returned by TrendSegment(——), (c) estimated signal with change-points returned by NOT (——) and ID (- - -), (d) estimated signal with change-points returned by TF (——) and CPOP (-·-·-).

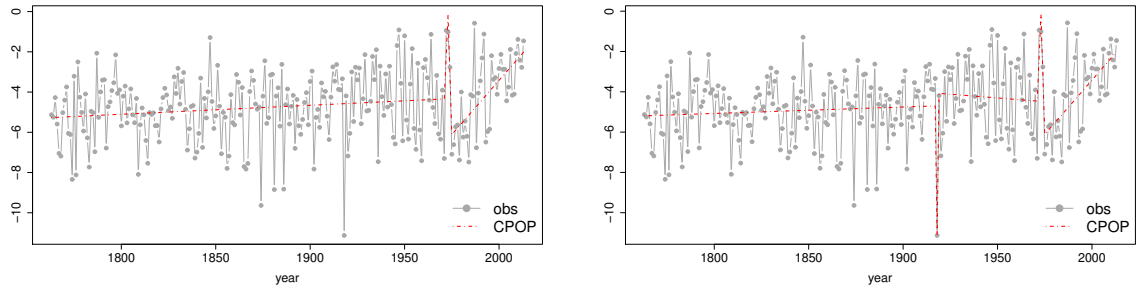
of our competing methods can estimate the 1918 observation as a single point by varying their tuning parameters; TF selects the optimal tuning parameter by minimising k-fold CV, thus we use a number of different values for k, but fail in finding an estimated fit that includes a point anomaly in 1918. ID requires a choice of constant for the threshold in a similar way that we need to choose an appropriate constant for the threshold. However, the number of estimated change-points increase suddenly (rather than gradually) with decreasing constant $\tilde{C}$ in the threshold $\lambda_{ID} = \tilde{C}\hat{\sigma}(2\log T)^{1/2}$. Figure 4.6 shows that ID reports no change-points when $\tilde{C} \in [0.335, 1.4]$, but suddenly detects so many change-points (including a single point in 1918) when $\tilde{C}$ decreases by 0.005 from $\tilde{C} = 0.335$ to $\tilde{C} = 0.330$. CPOP requires a choice of a parameter which penalises the number of estimated change-points (i.e. β in (2.27)), where the



(a) ID with $0.335 \leq \tilde{C} \leq 1.4$ (b) ID with $\tilde{C} = 0.330$

Fig. 4.6 Change-point analysis for January average temperature in Reykjavik from 1763 to 2013 in Section 4.5.1. (a) the data series (grey dots) and estimated signal with change-points returned by ID (- - -) when $0.335 \leq \tilde{C} \leq 1.4$ is used for the threshold $\lambda_{ID} = \tilde{C}\hat{\sigma}(2\log T)^{1/2}$, (b) when $\tilde{C} = 0.330$ is used for the threshold $\lambda_{ID} = \tilde{C}\hat{\sigma}(2\log T)^{1/2}$.

default is given as the Schwarz's Information Criterion (SIC, Schwarz (1978)) (i.e. $\beta = 2\log(n) = 11.07$ in this data example). We can control the value of β through the R function `CROPS.CPOP`, and Figure 4.7 shows the results under two different values of β; CPOP does not fit the 1918 observation as a single point when $\beta = 7.5$ but does so when $\beta = 5.5$.



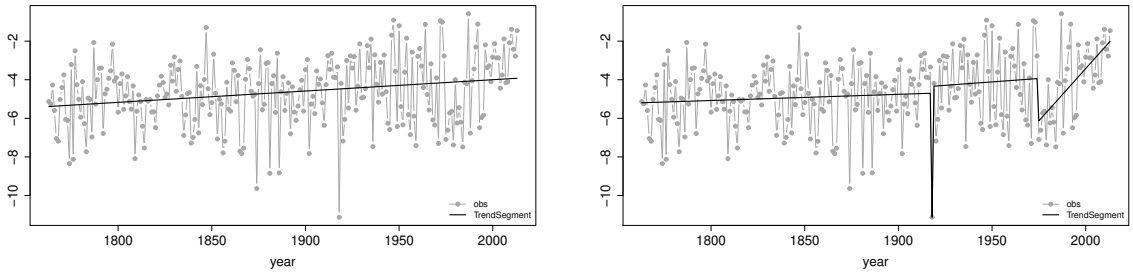
(a) CPOP with $\beta = 7.5$ (b) CPOP with $\beta = 5.5$

Fig. 4.7 Change-point analysis for January average temperature in Reykjavik from 1763 to 2013 in Section 4.5.1. (a) the data series (grey dots) and estimated signal with change-points returned by CPOP (-·-·-) when $\beta = 7.5$ is used, (b) when $\beta = 5.5$ is used.

To see how the choice of the threshold in the TrendSegment procedure affects the estimated change-points, we engage two different constants, $C = 1.5$ and $C = 1.0$,

for the threshold ($\lambda = C\sigma(2\log T)^{1/2}$) introduced in Section 4.4.1, where the default, $C = 1.3$, is used in Figure 4.5b. Figure 4.8 shows that the threshold with $C = 1.5$ returns no change-points while that with $C = 1.0$ detects one more change in 1974 compared to the estimated change-points when the default value ($C = 1.3$) is used. Interestingly, the added change-point, 1974, is the one reported by NOT and CPOP under their default parameter settings as shown in Figure 4.5.

This example illustrates the flexibility of the TrendSegment as it detects not only change-points in linear trend but it can identify a point anomaly at the same time, which the competing methods do not achieve under their default parameter settings.



(a) TrendSegment with $C = 1.5$ (b) TrendSegment with $C = 1.0$

Fig. 4.8 Change-point analysis for January average temperature in Reykjavik from 1763 to 2013 in Section 4.5.1. (a) the data series (grey dots) and estimated signal with change-points returned by TrendSegment(——) when $C = 1.5$ is used for the threshold $\lambda = C\hat{\sigma}(2\log T)^{1/2}$, (b) when $C = 1.0$ is used for the threshold $\lambda = C\hat{\sigma}(2\log T)^{1/2}$.

4.5.2 Monthly average sea ice extent of Arctic and Antarctic

We analyse the average sea ice extent of the Arctic and the Antarctic available from https://nsidc.org to estimate the change-points in its trend. As mentioned in Serreze and Meier (2018), sea ice extent is the most common measure for assessing the feature of high-latitude oceans and it is defined as the area covered with an ice concentration of at least 15%. Here we use the average ice extent in February and September as it is

known that the Arctic has the maximum ice extent typically in February while the minimum occurs in September and the Antarctic does the opposite.

Serreze and Meier (2018) indicate that the clear decreasing trend of sea ice extent of the Arctic in September is one of the most important indicators of climate change. In contrast to the Arctic, the sea ice extent of the Antarctic has been known to be stable in the sense that it shows a weak increasing trend in the decades preceding 2016 (Comiso et al., 2017; Serreze and Meier, 2018). However, Rintoul et al. (2018) warn of a possible collapse of the past stability by citing a significant decline of the sea ice extent in 2016. We now use the most up-to-date records (to 2018) and re-examine the concerns expressed in Rintoul et al. (2018) with the help of our change-point detection methodology.

Figures 4.9 and 4.10 show the well-known decreasing trend of the average sea ice extent in the Arctic both in its winter (February) and summer (September). In Figure 4.9, the TrendSegment estimate identifies change-points in 2004 and 2007 and detects a sudden drop during 2005-2007 which is also captured by TF and CPOP but ignored by NOT and ID. In Figure 4.10, TrendSegment and CPOP identify one change-point in 2006 which differentiates the decreasing speed of winter ice extent in the Arctic before and after 2006. The NOT estimate identifies two change-points where ID return a simple linear fit without any change-point.

As observed in the above-mentioned literature, the sea ice extent of the Antarctic shows a modest increasing trend up until recently (Figures 4.11 and 4.12); however, we observe a strong decreasing trend from the TrendSegment estimate with the detected change-point in 2016 for the Antarctic summer (February) and from 2015 for the Antarctic winter (September), which is in line with the message of Rintoul et al. (2018). Figure 4.11 shows that other methods also fit a strong decreasing trend by identifying

a change-point around 2014 in February of the Antarctic and Figure 4.12 shows similar results except that NOT returns no change-point.

The CAPA methodology proposed by Fisch et al. (2018) for detecting point anomalies and anomalous segments in terms of the model parameters (mean and variance) identifies one change-point in 2001 for ice extent of the Antarctic in February and report that both mean and variance increase after 2001. However, it has to be borne in mind that this methodology is designed for piecewise-constant, rather than piecewise-linear fits (whereas the data suggest that the latter may be more appropriate).

4.6 Proofs

4.6.1 Some useful lemmas

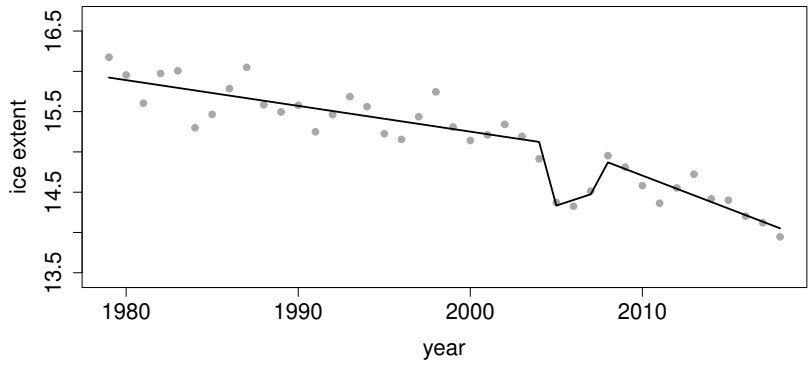
In this section, the proofs of Theorems 4.1-4.3 are given. We first present two preparatory lemmas.

Lemma 4.1 Let $\psi^{(j,k)} = \sum_{i=1}^{I^{(j,k)}} \phi_i^{(j,k)} g_i^{(j,k)}$ where $\phi_i^{(j,k)}$ are constants and $g_i^{(j,k)}$ are vectors of equal length with $\psi^{(j,k)}$ where $I^{(j,k)} \in \{3,4\}, j = 1,\ldots,J,\ k = 1,\ldots,K(j)$. If we define the set $G = \{g_l\}$ where there is a unique correspondence between $\left\{g_i^{(j,k)}{}_{i=1,\ldots,I^{(j,k)},j=1,\ldots,J,\,k=1,\ldots,K(j)}\right\}$ and $\{g_l\}$, we then have $P(A_T) \geq 1 - C_2 T^{-1}$ where

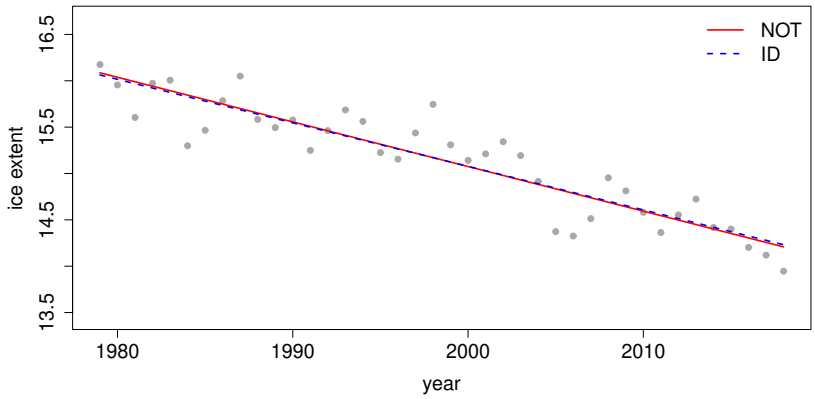
$$A_T = \left\{ \max_{g_l \in G} |g_l^\top \boldsymbol{\varepsilon}| \leq \lambda \right\}, \tag{4.40}$$

λ is as in Assumption 4.2 and C_2 is a positive constant.

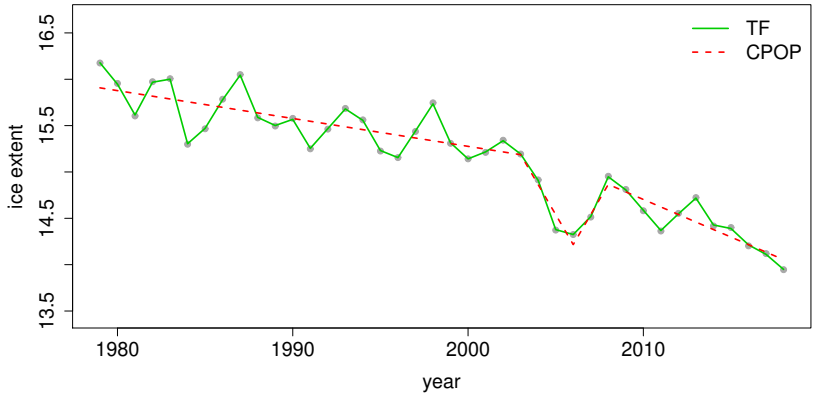
Proof. We firstly show that for any fixed (j,k), $g_i^{(j,k)}$ and $\phi_i^{(j,k)}$ satisfy the conditions, $\left(g_i^{(j,k)}\right)^\top g_i^{(j,k)} = 1$, $\left(g_i^{(j,k)}\right)^\top g_{i'}^{(j,k)} = 0$ and $\sum_i \left(\phi_i^{(j,k)}\right)^2 = 1$, where $\psi^{(j,k)} =$



(a) TrendSegment

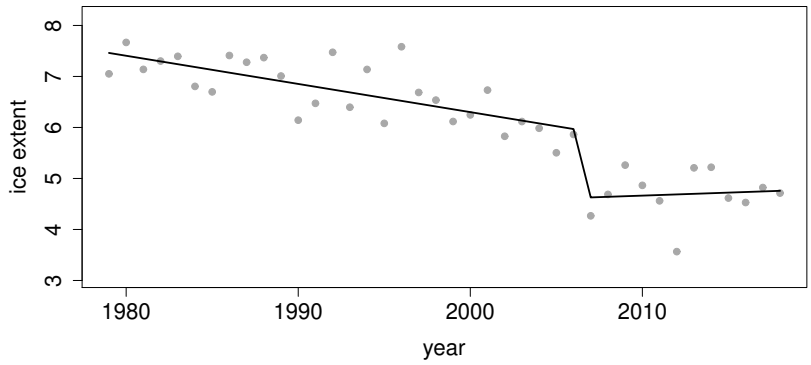


(b) NOT and ID

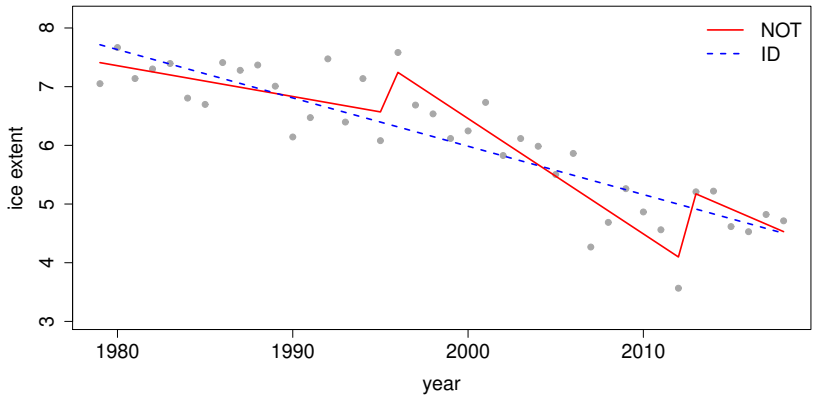


(c) TF and CPOP

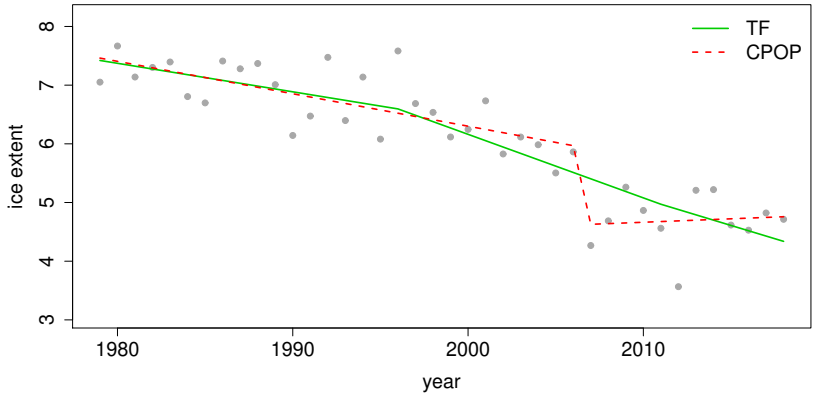
Fig. 4.9 Change-point analysis for the monthly average sea ice extent of the Arctic in February from 1979 to 2018 in Section 4.5.2. (a) the data series (grey dots) and the estimated signal with change-points returned by TrendSegment (——), (b) by NOT (——) and ID (- - -), (c) by TF (——) and CPOP (- - -).



(a) TrendSegment

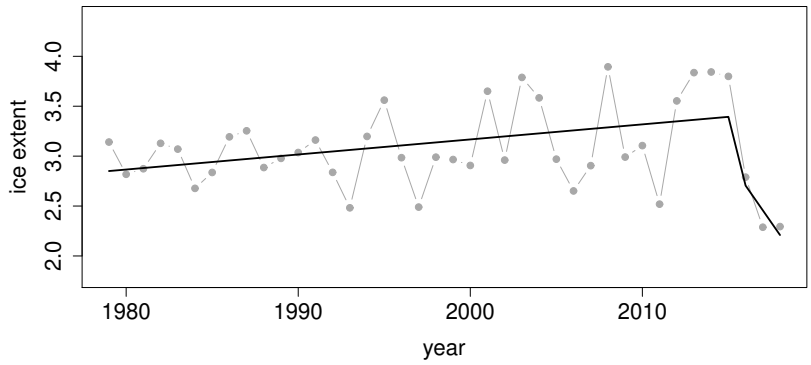


(b) NOT and ID

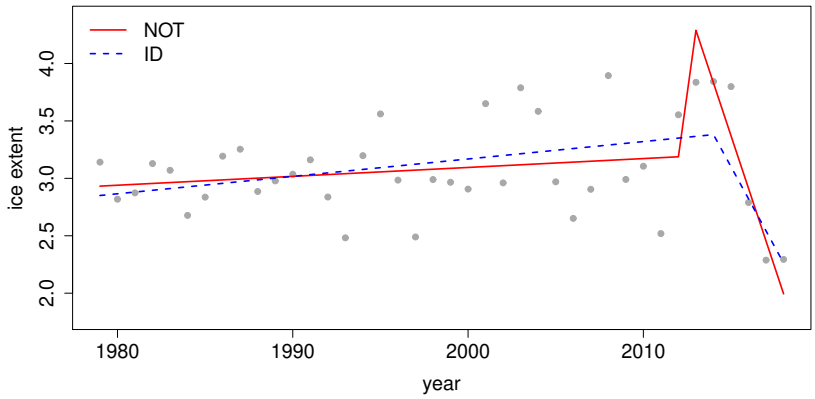


(c) TF and CPOP

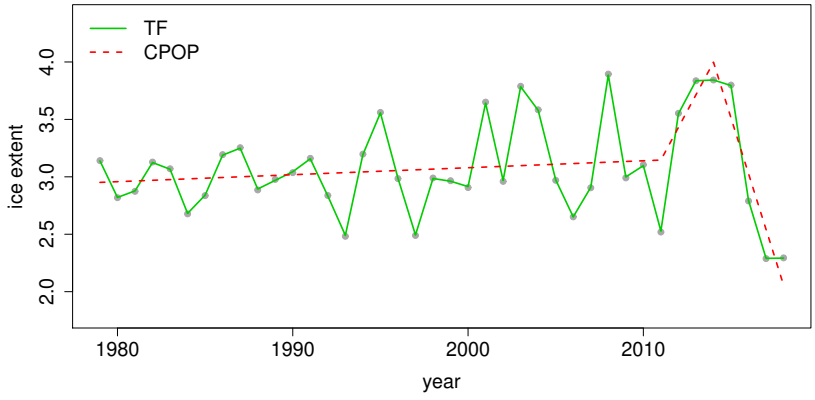
Fig. 4.10 Change-point analysis for the monthly average sea ice extent of the Arctic in September from 1979 to 2018 in Section 4.5.2. (a) the data series (grey dots) and the estimated signal with change-points returned by TrendSegment (——), (b) by NOT (——) and ID (- - -), (c) by TF (——) and CPOP (- - -).



(a) TrendSegment

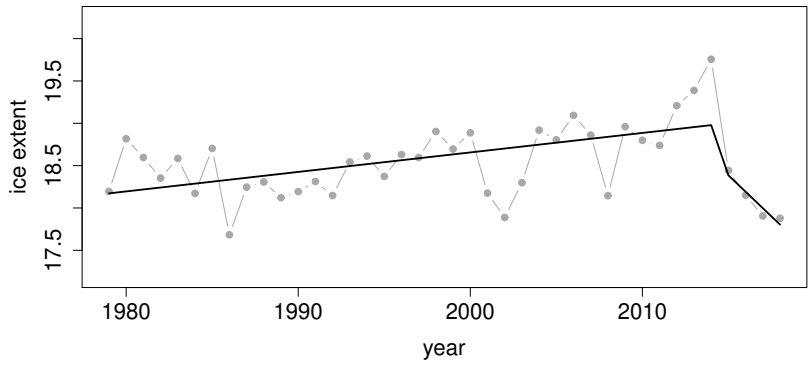


(b) NOT and ID

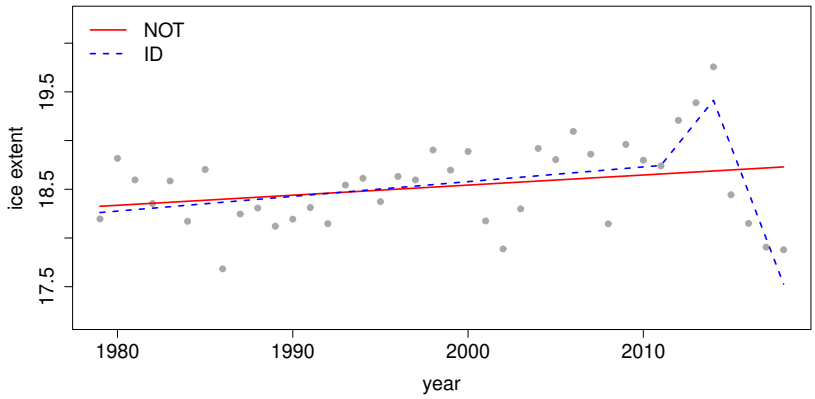


(c) TF and CPOP

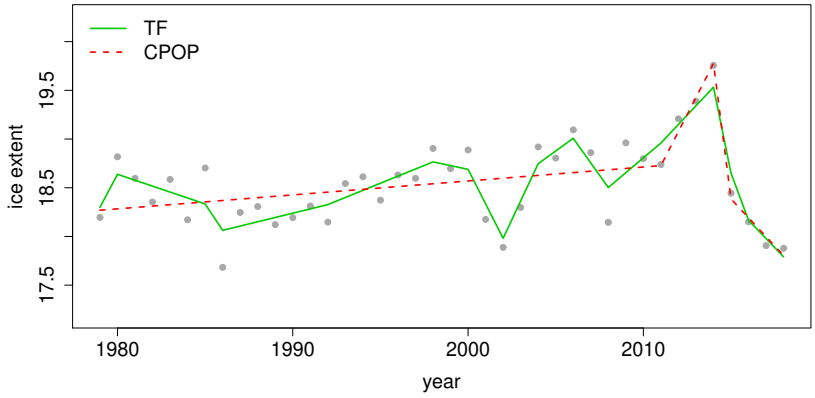
Fig. 4.11 Change-point analysis for the monthly average sea ice extent of the Antarctic in February from 1979 to 2018 in Section 4.5.2. (a) the data series (grey dots) and the estimated signal with change-points returned by TrendSegment (——), (b) by NOT (——) and ID (- - -), (c) by TF (——) and CPOP (- - -).



(a) TrendSegment



(b) NOT and ID



(c) TF and CPOP

Fig. 4.12 Change-point analysis for the monthly average sea ice extent of the Antarctic in September from 1979 to 2018 in Section 4.5.2. (a) the data series (grey dots) and the estimated signal with change-points returned by TrendSegment (——), (b) by NOT (——) and ID (- - -), (c) by TF (——) and CPOP (- - -).

$\sum_{i=1}^{I^{(j,k)}} \phi_i^{(j,k)} g_i^{(j,k)}$. Depending on the type of merge, $\psi^{(j,k)}$ fall into one of the followings,

$$
\begin{aligned}
\text{Type 1: } \psi_{p,q,r}^{(j,k)} &= \alpha_1 e_p + \alpha_2 e_{p+1} + \alpha_3 e_{p+2}, \\
\text{Type 2: } \psi_{p,q,r}^{(j,k)} &= \beta_1 e_p + \beta_2 (\underbrace{0,\ldots,0}_{p\times 1}, \boldsymbol{\ell}_{1,p+1,r}^\top, \underbrace{0,\ldots,0}_{(T-r)\times 1}) + \beta_3 (\underbrace{0,\ldots,0}_{p\times 1}, \boldsymbol{\ell}_{2,p+1,r}^\top, \underbrace{0,\ldots,0}_{(T-r)\times 1}), \\
\psi_{p,q,r}^{(j,k)} &= \beta_4 (\underbrace{0,\ldots,0}_{(p-1)\times 1}, \boldsymbol{\ell}_{1,p,r-1}^\top, \underbrace{0,\ldots,0}_{(T-r+1)\times 1}) + \beta_5 (\underbrace{0,\ldots,0}_{(p-1)\times 1}, \boldsymbol{\ell}_{2,p,r-1}^\top, \underbrace{0,\ldots,0}_{(T-r+1)\times 1}) + \beta_6 e_r, \\
\text{Type 3: } \psi_{p,q,r}^{(j,k)} &= \gamma_1 (\underbrace{0,\ldots,0}_{(p-1)\times 1}, \boldsymbol{\ell}_{1,p,q}^\top, \underbrace{0,\ldots,0}_{(T-q)\times 1}) + \gamma_2 (\underbrace{0,\ldots,0}_{(p-1)\times 1}, \boldsymbol{\ell}_{2,p,q}^\top, \underbrace{0,\ldots,0}_{(T-q)\times 1}) \\
&\quad + \gamma_3 (\underbrace{0,\ldots,0}_{q\times 1}, \boldsymbol{\ell}_{1,q+1,r}^\top, \underbrace{0,\ldots,0}_{(T-r)\times 1}) + \gamma_4 (\underbrace{0,\ldots,0}_{q\times 1}, \boldsymbol{\ell}_{2,q+1,r}^\top, \underbrace{0,\ldots,0}_{(T-r)\times 1}),
\end{aligned}
\tag{4.41}
$$

where e_i is a vector of length T having 1 only at i^{th} element and zero for the others. As is shown in Section 4.2.6, $\boldsymbol{\ell}_{1,i,j}$ and $\boldsymbol{\ell}_{2,i,j}$ are an arbitrary orthonormal basis of the subspace $\{(x_1, x_2, \ldots, x_{j-i+1}) \mid x_1 - x_2 = x_2 - x_3 = \cdots = x_{j-i} - x_{j-i+1}\}$ of $\mathbb{R}^{j-i+1}$.

In any case, we can obtain the representation $\psi^{(j,k)} = \sum_{i=1}^{I^{(j,k)}} \phi_i^{(j,k)} g_i^{(j,k)}$ from (4.41) if the constants $\phi_i^{(j,k)}$ correspond to $\{\alpha_i\}_{i=1}^3$ in Type 1, $\{\beta_i\}_{i=1}^3$ or $\{\beta_i\}_{i=4}^6$ in Type 2 and $\{\gamma_i\}_{i=1}^4$ in Type 3 and $g_i^{(j,k)}$ is the corresponding vector. From the orthonormality of the basis $(\boldsymbol{\ell}_{1,m,n}, \boldsymbol{\ell}_{2,m,n})$ for any (m,n), we see that the conditions, $\left(g_i^{(j,k)}\right)^\top g_i^{(j,k)} = 1$ and $\left(g_i^{(j,k)}\right)^\top g_{i'}^{(j,k)} = 0$, are satisfied for any (i, i', j, k) where $i \neq i'$. In addition, as $\psi^{(j,k)}$ keep orthonormality, we can argue that $\phi_i^{(j,k)}$ is bounded by the condition $\sum_i \left(\phi_i^{(j,k)}\right)^2 = 1$ for any (i, j, k) which implies $\sum_{i=1}^3 \alpha_i^2 = \sum_{i=1}^3 \beta_i^2 = \sum_{i=4}^6 \beta_i^2 = \sum_{i=1}^4 \gamma_i^2 = 1$ in (4.41).

If we predefine the pairs $(\boldsymbol{\ell}_{1,m,n}, \boldsymbol{\ell}_{2,m,n})$ for any (m,n) by choosing an orthonormal basis of the subspace $\{(x_1, x_2, \ldots, x_{n-m+1}) \mid x_1 - x_2 = x_2 - x_3 = \cdots = x_{n-m} - x_{n-m+1}\}$ of $\mathbb{R}^{n-m+1}$, then there exist at most T^2 vectors g_l in the set G. This is because m and n can be randomly chosen from $\{1, 2, \ldots, T\}$ with replacement and if $m \neq n$, the two drawn pairs, (m,n) and (n,m), correspond to the same basis vectors, $(\boldsymbol{\ell}_{1,m,n}, \boldsymbol{\ell}_{2,m,n})$,

while (m, m) correspond to one vector e_m. Now we are in position to show that $P(A_T) \geq 1 - C_2 T^{-1}$. Using a simple Bonferroni inequality, we have

$$1 - P(A_T) \leq \sum_G P(|Z| > \lambda) \leq 2T^2 \frac{\phi_Z(\lambda)}{\lambda} = \frac{1}{C_1 \sqrt{\pi} T^{C_1^2 - 2} \sqrt{\log T}} \leq \frac{C_2}{T} \tag{4.42}$$

where ϕ_Z is the p.d.f. of a standard normal Z,

$$P(|Z| > \lambda) = 2\frac{1}{\sqrt{2\pi}} \int_\lambda^\infty e^{-x^2/2} dx \leq 2\frac{1}{\sqrt{2\pi}} \int_\lambda^\infty \frac{x}{\lambda} e^{-x^2/2} dx = 2\frac{e^{-\lambda^2/2}}{\lambda\sqrt{2\pi}} \tag{4.43}$$

and

$$\frac{\phi_Z(\lambda)}{\lambda} = \frac{\frac{1}{\sqrt{2\pi}} e^{-C_1^2 \log T}}{C_1 \sqrt{2 \log T}} = \frac{1}{2C_1 \sqrt{\pi} \cdot T^{C_1^2} \sqrt{\log T}}. \tag{4.44}$$

This completes the proof.

Lemma 4.2 Let $\mathcal{S}_j^1 = \{1 \leq k \leq K(j) : d^{(j,k)}$ is $d_{p,q,r}$ such that $p < \eta_i + 1/2 < r$ for some $i = 1, \ldots, N\}$, and $\mathcal{S}_j^0 = \{1, \ldots, K(j)\} \setminus \mathcal{S}_j^1$. On the set A_T in (4.40) which satisfies $P(A_T) \to 1$ as $T \to \infty$, we have

$$\max_{\substack{j=1,\ldots,J, \\ k \in \mathcal{S}_j^0}} \left| d^{(j,k)} \right| \leq \lambda, \tag{4.45}$$

where λ is as in Assumption 4.2.

Proof. On the set A_T, the following holds for $j = 1, \ldots, J, k \in \mathcal{S}_j^0$,

$$\begin{aligned}
\left| d^{(j,k)} \right| &= \left| (\psi^{(j,k)})^\top \boldsymbol{\varepsilon} \right| \\
&= \left| \phi_1^{(j,k)} \left(g_1^{(j,k)} \right)^\top \boldsymbol{\varepsilon} + \phi_2^{(j,k)} \left(g_2^{(j,k)} \right)^\top \boldsymbol{\varepsilon} + \phi_3^{(j,k)} \left(g_3^{(j,k)} \right)^\top \boldsymbol{\varepsilon} + \phi_4^{(j,k)} \left(g_4^{(j,k)} \right)^\top \boldsymbol{\varepsilon} \right| \\
&\leq \max_{j,\, k} \left(\left| \phi_1^{(j,k)} \right| + \left| \phi_2^{(j,k)} \right| + \left| \phi_3^{(j,k)} \right| + \left| \phi_4^{(j,k)} \right| \right) \cdot \left(\max_{l:\, g_l \in G} \left| g_l^\top \boldsymbol{\varepsilon} \right| \right),
\end{aligned}$$

where $\boldsymbol{\varepsilon} = (\varepsilon_1, \ldots, \varepsilon_T)^\top$. The condition, $\sum_i \left(\phi_i^{(j,k)}\right)^2 = 1$ for any fixed (j,k), given in the proof of Lemma 4.1 implies that $\max_i \left|\phi_i^{(j,k)}\right| \leq 1$ for any (j,k), thus we have (4.45) when the constant C_1 for λ in (4.45) is larger than or equal to 4 times C_1 used in (4.40).

4.6.2 Proof of Theorems 4.1 - 4.3

Proof of Theorem 4.1 Let $\mathcal{S}_j^1$ and $\mathcal{S}_j^0$ as in Lemma 4.2. From the conditional orthonormality of the unbalanced wavelet transform, on the set A_T in (4.40), we have

$$\begin{aligned}
\|\tilde{\boldsymbol{f}} - \boldsymbol{f}\|_T^2 &= \frac{1}{T}\sum_{j=1}^{J}\sum_{k=1}^{K(j)}\left(d^{(j,k)} \cdot \mathbb{I}\Big\{\exists(j',k') \in \mathcal{C}_{j,k} \quad |d^{(j',k')}| > \lambda\Big\} - \mu^{(j,k)}\right)^2 \\
&\quad + T^{-1}\left(s_{1,T}^1 - \mu^{(0,1)}\right)^2 + T^{-1}\left(s_{1,T}^2 - \mu^{(0,2)}\right)^2 \\
&\leq \frac{1}{T}\sum_{j=1}^{J}\left(\sum_{k\in\mathcal{S}_j^0} + \sum_{k\in\mathcal{S}_j^1}\right)\left(d^{(j,k)} \cdot \mathbb{I}\Big\{\exists(j',k') \in \mathcal{C}_{j,k} \quad |d^{(j',k')}| > \lambda\Big\} - \mu^{(j,k)}\right)^2 \\
&\quad + 4C_1^2 T^{-1}\log T \\
&=: I + II + 4C_1^2 T^{-1}\log T, \qquad (4.46)
\end{aligned}$$

where $\mu^{(0,1)} = \langle \boldsymbol{f}, \psi^{(0,1)}\rangle$ and $\mu^{(0,2)} = \langle \boldsymbol{f}, \psi^{(0,2)}\rangle$. We note that $\left(s_{1,T}^1 - \mu^{(0,1)}\right)^2 \leq 2C_1^2\log T$ is simply obtained by combining Lemma 4.2 and the fact that $s_{1,T}^1 - \mu^{(0,1)} = \langle \boldsymbol{\varepsilon}, \psi^{(0,1)}\rangle$, which can also be applied to obtain $\left(s_{1,T}^2 - \mu^{(0,2)}\right)^2 \leq 2C_1^2\log T$. By Lemma 4.2, $\mathbb{I}\Big\{\exists(j',k') \in \mathcal{C}_{j,k} \quad |d^{(j',k')}| > \lambda\Big\} = 0$ for $k \in \mathcal{S}_j^0$ and also by the fact that $\mu^{(j,k)} = 0$ for $j = 1, \ldots, J, k \in \mathcal{S}_j^0$, we have $I = 0$. For II, we denote $\mathcal{B} = \Big\{\exists(j',k') \in$

$\mathcal{C}_{j,k} \quad |d^{(j',k')}| > \lambda \Big\}$ and have

$$\begin{aligned}\Big(d^{(j,k)} \cdot \mathbb{I}\{\mathcal{B}\} - \mu^{(j,k)}\Big)^2 &= \Big(d^{(j,k)} \cdot \mathbb{I}\{\mathcal{B}\} - d^{(j,k)} + d^{(j,k)} - \mu^{(j,k)}\Big)^2 \\ &\leq \Big(d^{(j,k)}\Big)^2 \mathbb{I}\Big(\Big|d^{(j',k')}\Big| \leq \lambda \text{ for all } (j',k') \in \mathcal{C}_{j,k}\Big) + \Big(d^{(j,k)} - \mu^{(j,k)}\Big)^2 \\ &\quad + 2\Big|d^{(j,k)}\Big| \, \mathbb{I}\Big(\Big|d^{(j',k')}\Big| \leq \lambda \text{ for all } (j',k') \in \mathcal{C}_{j,k}\Big) \, \Big|d^{(j,k)} - \mu^{(j,k)}\Big| \\ &\leq \lambda^2 + 2C_1^2 \log T + 2\lambda C_1 \{2 \log T\}^{1/2}. \end{aligned} \tag{4.47}$$

Combining with the upper bound of J, $\lceil \log(T)/\log((1-\rho)^{-1}) + \log(2)/\log(1-\rho) \rceil$, and the fact that $|\mathcal{S}_j^1| \leq N$, we have $II \leq 8C_1^2 N T^{-1} \lceil \log(T)/\log((1-\rho)^{-1}) + \log(2)/\log(1-\rho) \rceil \log T$, and therefore

$$\|\tilde{f} - f\|_T^2 \;\leq\; C_1^2 \; T^{-1} \; \log(T) \; \Big\{ 4 + 8N \; \lceil \log(T)/\log((1-\rho)^{-1}) + \log(2)/\log(1-\rho) \rceil \; \Big\}. \tag{4.48}$$

As the estimated change-points are obtained through those detail coefficients, thus at each scale, up to N estimated change-points are added. Combining it with the largest scale J whose order is $\log T$, the number of change-points in $\tilde{f}$ returned from the inverse TGUW transformation is up to $CN \log T$ where C is a constant.

Proof of Theorem 4.2 Let $\tilde{B}$ and $\tilde{\tilde{B}}$ the unbalanced wavelet basis corresponding to $\tilde{\boldsymbol{f}}$ and $\tilde{\tilde{\boldsymbol{f}}}$, respectively. As the change-points in $\tilde{\tilde{\boldsymbol{f}}}$ are a subset of those in $\tilde{\boldsymbol{f}}$, establishing $\tilde{\tilde{\boldsymbol{f}}}$ can be considered as applying the TGUW transform again to $\tilde{\boldsymbol{f}}$ which is just a repetition of procedure done in estimating $\tilde{\boldsymbol{f}}$ in the greediest way. Thus $\tilde{\tilde{B}}$ is classified into two categories, 1) all basis vectors $\psi^{(j,k)} \in \tilde{B}$ such that $\psi^{(j,k)}$ is not associated with the change-points in $\tilde{\boldsymbol{f}}$ and $|\langle \boldsymbol{X}, \psi^{(j,k)} \rangle| = |d^{(j,k)}| < \lambda$ and 2) all vectors $\psi^{(j,1)}$ produced in Stage 1 of post-processing.

We now investigate how many scales are used for this particular transform. First, the detail coefficients $d^{(j,k)}$ corresponding to the basis vectors $\psi^{(j,k)} \in \tilde{B}$ live on no

more than $J = O(\log T)$ scales and we have $|\mathcal{S}_j^1| \leq N$ by the argument used in the proof of Theorem 4.1. In addition, the vectors $\psi^{(j,1)}$ in the second category correspond to different change-points in $\tilde{\boldsymbol{f}}$ and there exist at most $\tilde{N} = O(N \log T)$ change-points in $\tilde{f}$ which we examine one at once (i.e. $|\mathcal{S}_j^1| \leq 1$), thus at most $\tilde{N}$ scales are required for $d^{(j,1)}$. Combining the results of two categories, the equivalent of quantity II in the proof of Theorem 4.1 for $\tilde{\tilde{\boldsymbol{f}}}$ is bounded by $II \leq C_3 N T^{-1} \log^2 T$ and this completes the proof of the l_2 result, $\left\|\tilde{\tilde{\boldsymbol{f}}} - \boldsymbol{f}\right\|_T^2 = O\Big(NT^{-1}\log^2(T)\Big)$ where C_3 is a positive constant large enough.

Finally, we show that there exist at most two change-points in $\tilde{\tilde{\boldsymbol{f}}}$ between true change-points $(\eta_\ell, \eta_{\ell+1})$ for $\ell = 0, \ldots, N$ where $\eta_0 = 0$ and $\eta_{N+1} = T$. Consider the case where three change-point for instance $(\tilde{\tilde{\eta}}_l, \tilde{\tilde{\eta}}_{l+1}, \tilde{\tilde{\eta}}_{l+2})$ lie between a pair of true change-point, $(\eta_\ell, \eta_{\ell+1})$. In this case, by Lemma 4.2, the maximum magnitude of two detail coefficients computed from the adjacent intervals, $[\tilde{\tilde{\eta}}_l + 1, \tilde{\tilde{\eta}}_{l+1}]$ and $[\tilde{\tilde{\eta}}_{l+1} + 1, \tilde{\tilde{\eta}}_{l+2}]$, is less than λ and $\tilde{\tilde{\eta}}_{l+1}$ would be get removed from the set of estimated change-points. This satisfies $\tilde{\tilde{N}} \leq 2(N+1)$.

Proof of Theorem 4.3 From Assumption 4.4, the followings hold.

- Given any $\epsilon > 0$ and $C > 0$, for some T_1 and all $T > T_1$, it holds that $\mathbb{P}\Big(\left\|\tilde{\tilde{\boldsymbol{f}}} - \boldsymbol{f}\right\|_T^2 > \frac{C^3}{4} R_T\Big) \leq \epsilon$ where $\tilde{\tilde{\boldsymbol{f}}}$ is the estimated signal specified in Theorem 4.2.

- For some T_2, and all $T > T_2$, it holds that $C^{1/3} T^{1/3} R_T^{1/3} (\underline{f}_T^\ell)^{-2/3} < \delta_T^\ell$ for all $\ell = 1, \ldots, N$.

Following the argument used in the proof of Theorem 19 in Lin et al. (2016), we take $T \geq T^*$ where $T^* = \max\{T_1, T_2\}$ and let $r_{\ell,T} = \lfloor C^{1/3} T^{1/3} R_T^{1/3} (\underline{f}_T^\ell)^{-2/3} \rfloor$ for $\ell = 1, \ldots, N$. Suppose that there exist at least one η_ℓ whose closest estimated change-point is not within the distance of $r_{\ell,T}$. Then there are no estimated change-points in

$\tilde{\tilde{\boldsymbol{f}}}$ within $r_{\ell,T}$ of η_ℓ which means that $\tilde{\tilde{f}}_j$ displays a linear trend over the entire segment $j \in \{\eta_\ell - r_{\ell,T}, \ldots, \eta_\ell + r_{\ell,T}\}$. Hence

$$\frac{1}{T}\sum_{j=\eta_\ell - r_{\ell,T}}^{\eta_\ell + r_{\ell,T}} \left(\tilde{\tilde{f}}_j - f_j\right)^2 \geq \frac{13 r_{\ell,T}^3}{24T}\left(\underline{f}_T^\ell\right)^2 > \frac{C^3}{4} R_T. \tag{4.49}$$

The first inequality holds by Lemma 20 of Lin et al. (2016), and the second one holds by the definition of $r_{\ell,T}$. Assuming that at least one η_ℓ does not have an estimated change-point within the distance of $r_{\ell,T}$ implies that the estimation error exceeds $\frac{C^3}{4} R_T$ which is a contradiction as it is an event that we know occurs with probability at most ϵ. Therefore, there must exist at least one estimated change-point within the distance of $r_{\ell,T}$ from each true change point η_ℓ.

Throughout Stage 2 of post-processing, $\tilde{\tilde{\eta}}_{\ell_0}$ is either the closest estimated change-point of any η_ℓ or not. If $\tilde{\tilde{\eta}}_{\ell_0}$ is not the closest estimated change-point to the nearest true change-point on either its left or its right, by the construction of detail coefficients in Stage 2 of post-processing, Lemma 4.2 guarantees that the corresponding detail coefficient has the magnitude less than λ and $\tilde{\tilde{\eta}}_{\ell_0}$ gets removed. Suppose $\tilde{\tilde{\eta}}_{\ell_0}$ is the closest estimated change-point of a true change-point η_ℓ and it is within the distance of $CT^{1/3}R_T^{1/3}\left(\underline{f}_T^\ell\right)^{-2/3}$ from η_ℓ. If the corresponding detail coefficient has the magnitude less than λ and $\tilde{\tilde{\eta}}_{\ell_0}$ is removed, there must exist another $\tilde{\tilde{\eta}}_\ell$ within the distance of $CT^{1/3}R_T^{1/3}\left(\underline{f}_T^\ell\right)^{-2/3}$ from η_ℓ. If there are no such $\tilde{\tilde{\eta}}_\ell$, then by the construction of the detail coefficient, the order of magnitude of $\left|d_{p_{\ell_0},q_{\ell_0},r_{\ell_0}}\right|$ would be such that $\left|d_{p_{\ell_0},q_{\ell_0},r_{\ell_0}}\right| > \lambda$ thus $\tilde{\tilde{\eta}}_{\ell_0}$ would not get removed. Therefore, after Stage 2 of post-processing is finished, each true change-point η_ℓ has its unique estimator within the distance of $CT^{1/3}R_T^{1/3}\left(\underline{f}_T^\ell\right)^{-2/3}$.

Chapter 5

Trend Segmentation for high-dimensional panel data

5.1 Introduction

In this chapter, we consider one panel of n univariate data sequences where the dimension n and the length of data sequences T may be large and the dimension is comparable with, or even larger than the length of data sequences. We propose the change-point model for high-dimensional panel data,

$$X_{i,t} = f_{i,t} + \varepsilon_{i,t}, \quad i = 1, \ldots, n, \ t = 1, \ldots, T \tag{5.1}$$

where $\boldsymbol{f}_i = (f_{i,1}, \ldots, f_{i,T})^\top$ is the underlying signal vector of the observation $\boldsymbol{X}_i = (X_{i,1}, \ldots, X_{i,T})^\top$. For each i, $\boldsymbol{\varepsilon}_i = (\varepsilon_{i,1}, \ldots, \varepsilon_{i,T})^\top$ is the independent Gaussian random error with the conditions that $E(\boldsymbol{\varepsilon}_i) = 0$, $\mathrm{Var}(\boldsymbol{\varepsilon}_i) = \sigma_i^2$. The errors can be dependent across the panel. Including this case, in Section 5.3, we explore two other cases, 1) when the errors have temporal dependence and 2) when the cross-sectional dependence is captured through a specific structure.

We assume that the signal vectors $\{\boldsymbol{f}_i\}_{i=1}^n$ share N distinct change-points,

$$0 = \eta_0 < \eta_1 < \eta_2 < \ldots < \eta_N < \eta_{N+1} = T, \tag{5.2}$$

in that at each change-point η_ℓ, there exists at least one signal $\boldsymbol{f}_i$ in which the trend change occurs at f_{i,η_ℓ}. For each change-point η_ℓ, the change can occur in a dense subset of the coordinates (e.g. all coordinates $\{\boldsymbol{f}_i\}_{i=1,\ldots,n}$) or only in a sparse subset of the coordinates, where the sparsity is formulated later in Section 5.2.1. The signal vectors, $\{\boldsymbol{f}_i\}_{i=1}^n$, are assumed to have a form of either piecewise-constant or piecewise-linear between any adjacent change-points, η_ℓ and $\eta_{\ell+1}$. The piecewise-linear signal does not need to be continuous at change-points and this will be formulated later in Section 5.2.1. The value of N is unknown and can grow with n and T.

We introduce a new methodology invented for multiple trend change detection in high-dimensional panel data which we refer to as High-dimensional Trend Segmentation (HiTS) in this chapter. HiTS performs well for a set of signals with long trend segments or frequent change-points with short segments or a mix of those. Besides, it is designed to work well in a particular setting where only a sparse subset of coordinates have changes in trend. The main ingredient of the HiTS procedure is a new High-dimensional Tail-Greedy Unbalanced Wavelet (HiTGUW) transform that is a conditionally orthonormal, bottom-up transform for high-dimensional panel data through an adaptively constructed unbalanced wavelet basis. The HiTGUW transform is an extension of the TGUW transform introduced in Chapter 4 for a univariate data sequence into high-dimensional settings. As in the case of TGUW, the HiTGUW transform is also achieved in a data-driven way in that a wavelet basis is constructed through recursively aggregating the information of all coordinates. In Section 5.3, the HiTS algorithm is shown to be statistically consistent in estimating the number and the locations of change-points and in Sections 5.4 and 5.5, we show that HiTS provides

a good performance not only in the case where the changes in trend occur in most of the coordinates but also when only a sparse subset of data sequences undergoes the changes. Other benefits of the HiTS procedure include low computational cost and ease of implementation.

Change-point analysis for high-dimensional time series has recently received much attention in the literature. Many of the existing works study the case when $\{\boldsymbol{f}_i\}_{i=1}^n$ in (5.1) are modelled as piecewise-constant and a review of the relevant literature can be found in Section 2.3.3. This is an important applied problem in a variety of fields, for example when we have a land temperature dataset that consists of average temperatures recorded in 50 cities of South Africa for 157 years as shown in Figure 5.1, our interest is in detecting and locating change-points in time that are shared by 50 cities in the way we define in (5.2) and if any change-point is detected, it is also of interest to find cities in which the estimated change-point is truly located. Both directions are explored in Section 5.5.1. Importantly, the temperature curves in Figure 5.1 appear to have cross-sectional dependence in that all curves tend to move together depending on years. The asymptotic behaviour of the estimated change-points under cross-sectional dependence is explored in Section 5.3.

The other commonly-encountered signals in practice include the piecewise-linear structure. Investigating common changes in piecewise-linear panel data is an important task as the simplest model designed for detecting level changes cannot give any useful information when the underlying signal has a form of piecewise-linear or when the interest is in detecting change-points in slope. Despite the simplicity of the concept, to the best of our knowledge, detecting multiple change-points in linear trend for high-dimensional panel data has not previously been studied. The HiTS algorithm introduces a new way of detecting multiple change-points in both piecewise-constant and piecewise-linear trends, which can in principle be extended to higher-order polynomials,

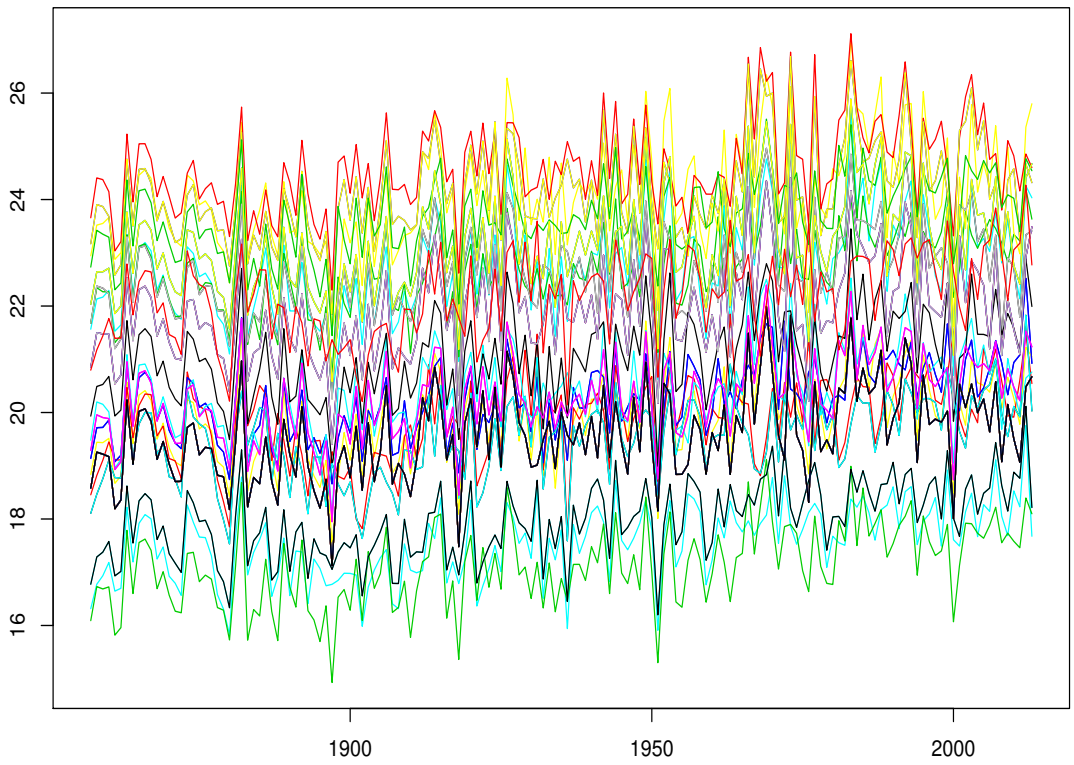


Fig. 5.1 January average temperature curves of 50 cities in South Africa from 1857 to 2013.

but we do not pursue in this work. In Section 5.5.2, the usefulness of the HiTS algorithm in detecting multiple changes in linear trend is illustrated through a climate data that consists of monthly average sea ice extent of Arctic and Antarctic.

Considering the previous works for detecting multiple change-points for high-dimensional panel data, many of them are heavily inspired by Binary Segmentation (BS, Vostrikova (1981)) and its variants e.g. wild binary segmentation (Fryzlewicz, 2014). However, as shown in Maeng and Fryzlewicz (2019) and Fryzlewicz (2018b) under the univariate data setting, the top-down (i.e. divisive) approaches such as BS often fail to perform adequately in the case of multiple change-points whereas the bottom-up (i.e. agglomerative) procedure which recursively merges the neighbouring regions of the data performs better. The current work extends the bottom-up procedures studied

in Fryzlewicz (2018b) and Maeng and Fryzlewicz (2019) to the problem of detecting changes in mean and in slope (respectively) in high-dimensional panel data. However, we emphasise that the HiTS procedure has entirely different goals from those two methods designed for univariate data sequences. The HiTS algorithm focuses on the aggregation of the adaptively-obtained statistics from the high-dimensional panel data where the details will be specified later in Section 5.2.3. Importantly, the aggregation is designed to work well in the extremely sparse case when a very small number of coordinates change at some change-points. As will be shown in Section 5.4, HiTS works substantially better than existing competitors in estimating multiple change-points when the signal is cross-sectionally extremely sparse and/or when long, short or a mix of those trend segments exist.

The outline of the remainder of this chapter is as follows. In Section 5.2, we give a full description of the HiTS procedure under two scenarios and Section 5.3 presents the relevant theoretical results under various assumptions on the errors. The supporting numerical studies are given in Section 5.4 and the usefulness of our methodology is illustrated in Section 5.5 through South Africa temperature data and sea ice extent data. The proofs of our main theoretical results are in Section 5.6.

5.2 Methodology

5.2.1 Settings

The following two commonly-encountered scenarios are investigated in this work.

(S1) Piecewise-constant structure:

$$f_{i,t} = \theta_{i,\ell} \quad \text{for} \quad t \in [\eta_{\ell-1}+1, \eta_\ell], \ \ell = 1, \ldots, N+1, \tag{5.3}$$
$$\text{where } \exists\, \Omega_\ell = \Big\{ i : \big| f_{i,\eta_\ell+1} - f_{i,\eta_\ell} \big| \neq 0 \Big\} \subset \Big\{ 1, \ldots, n \Big\} \text{ such that } \Omega_\ell \neq \emptyset \text{ for } \ell = 1, \ldots, N.$$

(S2) Piecewise-linear structure:

$$f_{i,t} = \theta^1_{i,\ell} + \theta^2_{i,\ell}\, t \quad \text{for} \quad t \in [\eta_{\ell-1}+1, \eta_\ell],\ \ell = 1, \ldots, N+1, \tag{5.4}$$
$$\text{where } \exists\ \Omega_\ell = \left\{i : f_{i,\eta_\ell} + \theta^2_{i,\ell} \neq f_{i,\eta_\ell+1}\right\} \subset \left\{1, \ldots, n\right\} \text{ such that } \Omega_\ell \neq \emptyset \text{ for } \ell = 1, \ldots, N.$$

We note that $f_{i,t}$ is the underlying signal in model (5.1). The definition of (S2) permits both continuous and discontinuous changes.

5.2.2 Structure of HiTS

The skeleton of the HiTS procedure for estimating the number and the locations of change-points is similar to that of TrendSegment in Chapter 4 and consists of the following four steps.

1. *HiTGUW transformation.* Perform the HiTGUW transform to the input data matrix by recursively applying the conditionally orthonormal transformations to the same local regions of all vectors $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$ in a bottom-up way. This is an unbalanced adaptive wavelet transformation and produces a data-adaptive multiscale decomposition of the data matrix with detail-type coefficients of the dimension $n \times (T-1)$ and smooth coefficients of the dimension $n \times 1$ in scenario (S1), and with detail-type coefficients of the dimension $n \times (T-2)$ and smooth coefficients of the dimension $n \times 2$ in scenario (S2). The novelty of this transformation comes from the way of aggregating detail-type coefficients that decide which regions should be merged first. The details can be found in Section 5.2.3.
2. *Thresholding.* Aggregate the detail coefficients coordinate-wisely. If the magnitude of the (aggregated) detail coefficients is smaller than a pre-specified threshold then set to zero those of corresponding (non-aggregated) detail coefficients as

long as all the non-zero (aggregated) detail coefficients are connected to each other in the tree structure which shows the merging history. This step decides the significance of the sparse representation suggested in the HiTGUW transformation.

3. *Inverse HiTGUW transformation.* Carry out the inverse HiTGUW transformation with the thresholded coefficients in step 2 and this gives initial estimates of $\boldsymbol{f}_1, \ldots, \boldsymbol{f}_n$ that can be shown to be l_2-consistent, but not yet consistent for the number and the locations of change-points.
4. *Post-processing.* Perform the two stages of post-processing by removing some change-points shown to be spurious. This step enables us to achieve estimation consistency for the number and the locations of change-points.

In the following four sections, we describe each step above in order for both scenarios (S1) and (S2) given in (5.3) and (5.4), respectively.

5.2.3 HiTGUW transformation

In this section, we describe the HiTGUW transformation in detail. We first provide a simple example of the HiTGUW transformation in each scenario to help readers understand the entire procedures at a glance and then formulate the HiTGUW transformation in generality in each scenario. In the initial stage, the input data is considered smooth coefficients and the HiTGUW transform iteratively updates the sequences of smooth coefficients by merging the adjacent sections of the data which are most likely to belong to the same segment in terms of the polynomial trend of interest. We emphasise that at each merge, the same sections of the coordinates are merged at once and those sections are chosen by aggregating the features of all coordinates. The following examples show single merges at each pass through the data, but we will later generalise it to multiple passes through the data, which speed up computation where

the device is called "tail-greediness" as is in Chapter 4. We refer to j^{th} pass through the data as scale j. We note that mergings performed in scenario (S1) have no particular type but merges in scenario (S2) can be classified into one of three forms, Type 1, 2 and 3, where Type 2 and 3 merges are built under the "two together" rule introduced in Section 4.2.2. The notation for the following examples and for the general algorithm introduced later is in Table 5.1.

Table 5.1 Notation. See Section 5.2.3 for formulae for the terms listed.

$\boldsymbol{X}_i = (X_{i,1}, \ldots, X_{i,T})$	i^{th} data sequence.
$\boldsymbol{S}$	data sequence matrix of the dimension $n \times T$ containing the (recursively updated) smooth and detail coefficients from the initial input of $\boldsymbol{S}^0$.
$S^0_{i,t}$	the element of the matrix $\boldsymbol{S}^0$ where $S^0_{i,t} = X_{i,t}$.
$d_{i,[p,q,r]}$	detail coefficient obtained from $\{X_{i,p}, \ldots, X_{i,r}\}$ (all merges in scenario (S1) and merges of Types 1 or 2 in scenario (S2)).
$d^1_{i,[p,q,r]}, d^2_{i,[p,q,r]}$	paired detail coefficients obtained by merging two adjacent subintervals, $\{X_{i,p}, \ldots, X_{i,q}\}$ and $\{X_{i,q+1}, \ldots, X_{i,r}\}$, where $r > q+2$ and $q > p+1$ (merge of Type 3 in scenario (S2)).
$s_{i,[p,r]}$	smooth coefficients obtained from $\{X_{i,p}, \ldots, X_{i,r}\}$ in scenario (S1).
$s^1_{i,[p,r]}, s^2_{i,[p,r]}$	smooth coefficients obtained from $\{X_{i,p}, \ldots, X_{i,r}\}$, paired under the "two together" rule in scenario (S2).

Example for scenario (S1)

We provide a simple example of the HiTGUW transformation in scenario (S1) where the accompanying illustration can be found in Figure 5.2. Assume that we have the initial input data matrix of the dimension 3×5,

$$\boldsymbol{S}^0 = \begin{pmatrix} X_{1,1} & X_{1,2} & X_{1,3} & X_{1,4} & X_{1,5} \\ X_{2,1} & X_{2,2} & X_{2,3} & X_{2,4} & X_{2,5} \\ X_{3,1} & X_{3,2} & X_{3,3} & X_{3,4} & X_{3,5} \end{pmatrix}, \tag{5.5}$$

thus the complete algorithm consists of 4 merges.

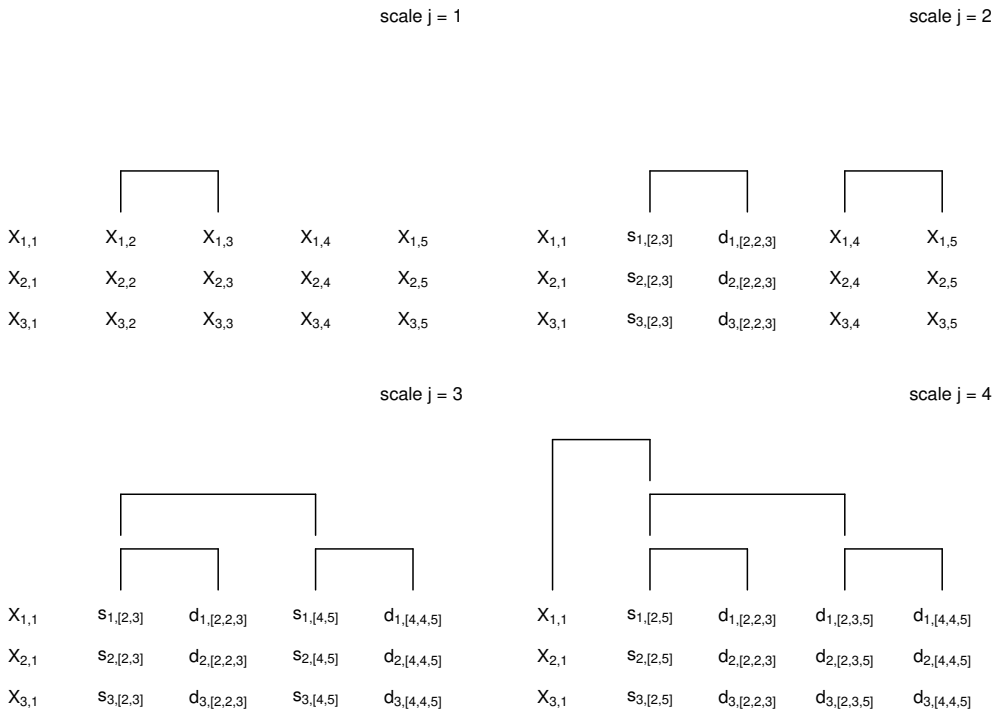


Fig. 5.2 Construction of tree for the example of scenario (S1) in Section 5.2.3; each diagram shows all merges performed up to the given scale.

Scale $j = 1$. From the initial input $\boldsymbol{S}^0$ in (5.5), we consider 4 pairs of columns, $(1^{\text{st}}, 2^{\text{nd}})$, $(2^{\text{nd}}, 3^{\text{rd}})$, $(3^{\text{rd}}, 4^{\text{th}})$, $(4^{\text{th}}, 5^{\text{th}})$, compute the detail vector for each pair of columns (where the formula can be found in (5.16)) and obtain the aggregated detail coefficient for each detail vector (from the formula, $d^*_{[p,q,r]} = \max_i |d_{i,[p,q,r]}|$) as follows:

$$\begin{bmatrix} X_{1,1} & X_{1,2} \\ X_{2,1} & X_{2,2} \\ X_{3,1} & X_{3,2} \end{bmatrix} \to \begin{bmatrix} d_{1,[1,1,2]} \\ d_{2,[1,1,2]} \\ d_{3,[1,1,2]} \end{bmatrix} \to d^*_{[1,1,2]}, \quad \begin{bmatrix} X_{1,2} & X_{1,3} \\ X_{2,2} & X_{2,3} \\ X_{3,2} & X_{3,3} \end{bmatrix} \to \begin{bmatrix} d_{1,[2,2,3]} \\ d_{2,[2,2,3]} \\ d_{3,[2,2,3]} \end{bmatrix} \to d^*_{[2,2,3]},$$

$$\begin{bmatrix} X_{1,3} & X_{1,4} \\ X_{2,3} & X_{2,4} \\ X_{3,3} & X_{3,4} \end{bmatrix} \to \begin{bmatrix} d_{1,[3,3,4]} \\ d_{2,[3,3,4]} \\ d_{3,[3,3,4]} \end{bmatrix} \to d^*_{[3,3,4]}, \quad \begin{bmatrix} X_{1,4} & X_{1,5} \\ X_{2,4} & X_{2,5} \\ X_{3,4} & X_{3,5} \end{bmatrix} \to \begin{bmatrix} d_{1,[4,4,5]} \\ d_{2,[4,4,5]} \\ d_{3,[4,4,5]} \end{bmatrix} \to d^*_{[4,4,5]},$$

where the absolute values of the aggregated details are compared. Suppose that $d^*_{[2,2,3]}$ has the smallest size, then merge the corresponding pair of columns and update the initial input matrix in (5.5) into the following data sequence matrix:

$$\boldsymbol{S} = \begin{pmatrix} X_{1,1} & s_{1,[2,3]} & d_{1,[2,2,3]} & X_{1,4} & X_{1,5} \\ X_{2,1} & s_{2,[2,3]} & d_{2,[2,2,3]} & X_{2,4} & X_{2,5} \\ X_{3,1} & s_{3,[2,3]} & d_{3,[2,2,3]} & X_{3,4} & X_{3,5} \end{pmatrix}. \tag{5.6}$$

As will be specified later, the l_∞-aggregation of the detail coefficients enables the HiTGUW transform to provide a good performance not only when the changes occur in most of the data sequences but also when only a sparse subset of data sequences undergoes the changes.

Scale $j = 2$. From now on, we ignore any detail coefficient columns in the updated data matrix. Then the possible pairs of neighbouring columns for next merging are:

$$\begin{bmatrix} X_{1,1} & s_{1,[2,3]} \\ X_{2,1} & s_{2,[2,3]} \\ X_{3,1} & s_{3,[2,3]} \end{bmatrix}, \begin{bmatrix} s_{1,[2,3]} & X_{1,4} \\ s_{2,[2,3]} & X_{2,4} \\ s_{3,[2,3]} & X_{3,4} \end{bmatrix}, \begin{bmatrix} X_{1,4} & X_{1,5} \\ X_{2,4} & X_{2,5} \\ X_{3,4} & X_{3,5} \end{bmatrix},$$

where their corresponding aggregated detail coefficients are $d^*_{[1,1,3]}$, $d^*_{[2,3,4]}$, $d^*_{[4,4,5]}$, respectively. Assume that the last pair of columns gives the smallest size of the aggregated detail coefficient among 3 candidates, then we merge them through the orthogonal transformation formulated in (5.18). The data matrix is now updated into

$$\boldsymbol{S} = \begin{pmatrix} X_{1,1} & s_{1,[2,3]} & d_{1,[2,2,3]} & s_{1,[4,5]} & d_{1,[4,4,5]} \\ X_{2,1} & s_{2,[2,3]} & d_{2,[2,2,3]} & s_{2,[4,5]} & d_{2,[4,4,5]} \\ X_{3,1} & s_{3,[2,3]} & d_{3,[2,2,3]} & s_{3,[4,5]} & d_{3,[4,4,5]} \end{pmatrix}. \tag{5.7}$$

Scale $j = 3$. We now compare the following two candidates for merging,

$$\begin{bmatrix} X_{1,1} & s_{1,[2,3]} \\ X_{2,1} & s_{2,[2,3]} \\ X_{3,1} & s_{3,[2,3]} \end{bmatrix}, \begin{bmatrix} s_{1,[2,3]} & s_{1,[4,5]} \\ s_{2,[2,3]} & s_{2,[4,5]} \\ s_{3,[2,3]} & s_{3,[4,5]} \end{bmatrix}.$$

Suppose that the second merging is preferred, then we update the data sequence into

$$\boldsymbol{S} = \begin{pmatrix} X_{1,1} & s_{1,[2,5]} & d_{1,[2,2,3]} & d_{1,[2,3,5]} & d_{1,[4,4,5]} \\ X_{2,1} & s_{2,[2,5]} & d_{2,[2,2,3]} & d_{2,[2,3,5]} & d_{2,[4,4,5]} \\ X_{3,1} & s_{3,[2,5]} & d_{3,[2,2,3]} & d_{3,[2,3,5]} & d_{3,[4,4,5]} \end{pmatrix}, \tag{5.8}$$

by performing an orthonormal transformation.

Scale $j = 4$. We have only one pair of columns available:

$$\begin{bmatrix} X_{1,1} & s_{1,[2,5]} \\ X_{2,1} & s_{2,[2,5]} \\ X_{3,1} & s_{3,[2,5]} \end{bmatrix},$$

and the orthonormal transformation gives the following updated data matrix,

$$\boldsymbol{S} = \begin{pmatrix} s_{1,[1,5]} & d_{1,[1,1,5]} & d_{1,[2,2,3]} & d_{1,[2,3,5]} & d_{1,[4,4,5]} \\ s_{2,[1,5]} & d_{2,[1,1,5]} & d_{2,[2,2,3]} & d_{2,[2,3,5]} & d_{2,[4,4,5]} \\ s_{3,[1,5]} & d_{3,[1,1,5]} & d_{3,[2,2,3]} & d_{3,[2,3,5]} & d_{3,[4,4,5]} \end{pmatrix}. \tag{5.9}$$

Therefore, the transformation is completed with $T - 1 = 4$ columns of detail coefficients and 1 column of smooth coefficients.

Example for scenario (S2)

Unlike scenario (S1), the HiTGUW algorithm for the piecewise-linear signals in scenario (S2) requires the high-dimensional version of the "two together" rule that forces any paired smooth coefficient vectors returned by an orthonormal transform not to be separated in any subsequent merges. This is a natural requirement as any such paired smooth coefficient vectors contain information about local linear regression fits.

In addition, only in scenario (S2), as merging continues under the high-dimensional version of the "two together" rule, all merges can have one of three forms, Type 1: merging three initial smooth coefficient vectors, Type 2: merging one initial smooth coefficient vector and a paired vectors of smooth coefficients and Type 3: merging two sets of (paired) vectors of smooth coefficients which is composed of two merges of triplets of smooth coefficient vectors. The following example demonstrates all three possible types of merges.

We now provide a simple example of the HiTGUW transformation in scenario (S2), which produces a tree whose structure is the same as the one constructed in the previous example in Section 4.2.2. The length of the panel data used in this example is also the same as that of the data sequence in Section 4.2.2, but the dimension is different; now we consider three univariate data sequences (i.e. the dimensionality is $n = 3$), while a univariate data sequence (i.e. $n = 1$) is considered in Section 4.2.2. The accompanying illustration is in Figure 5.3 and the relevant notation can be found in Table 5.1. Assume that we have the initial input matrix of the dimension 3×8,

$$\boldsymbol{S}^0 = \begin{pmatrix} X_{1,1} & X_{1,2} & X_{1,3} & X_{1,4} & X_{1,5} & X_{1,6} & X_{1,7} & X_{1,8} \\ X_{2,1} & X_{2,2} & X_{2,3} & X_{2,4} & X_{2,5} & X_{2,6} & X_{2,7} & X_{2,8} \\ X_{3,1} & X_{3,2} & X_{3,3} & X_{3,4} & X_{3,5} & X_{3,6} & X_{3,7} & X_{3,8} \end{pmatrix}, \tag{5.10}$$

so that the complete HiTGUW transform consists of 6 merges. We show 6 example merges one by one under the high-dimensional "two together" rule.

Scale $j = 1$. From the initial input $\boldsymbol{S}^0$ in (5.10), we consider 6 triplets of columns, $(1^{\text{st}}, 2^{\text{nd}}, 3^{\text{rd}})$, $(2^{\text{nd}}, 3^{\text{rd}}, 4^{\text{th}})$, $(3^{\text{rd}}, 4^{\text{th}}, 5^{\text{th}})$, $(4^{\text{th}}, 5^{\text{th}}, 6^{\text{th}})$, $(5^{\text{th}}, 6^{\text{th}}, 7^{\text{th}})$, $(6^{\text{th}}, 7^{\text{th}}, 8^{\text{th}})$, and compute the detail vector for each triplet of columns (where the formula can be found in (5.19)) and obtain the aggregated detail coefficient for each detail vector (from the formula, $d^*_{[p,q,r]} = \max_i |d_{i,[p,q,r]}|$) as follows:

$$\begin{bmatrix} X_{1,1} & X_{1,2} & X_{1,3} \\ X_{2,1} & X_{2,2} & X_{2,3} \\ X_{3,1} & X_{3,2} & X_{3,3} \end{bmatrix} \rightarrow \begin{bmatrix} d_{1,[1,2,3]} \\ d_{2,[1,2,3]} \\ d_{3,[1,2,3]} \end{bmatrix} \rightarrow d^*_{[1,2,3]}, \quad \begin{bmatrix} X_{1,2} & X_{1,3} & X_{1,4} \\ X_{2,2} & X_{2,3} & X_{2,4} \\ X_{3,2} & X_{3,3} & X_{3,4} \end{bmatrix} \rightarrow \begin{bmatrix} d_{1,[2,3,4]} \\ d_{2,[2,3,4]} \\ d_{3,[2,3,4]} \end{bmatrix} \rightarrow d^*_{[2,3,4]},$$

$$\begin{bmatrix} X_{1,3} & X_{1,4} & X_{1,5} \\ X_{2,3} & X_{2,4} & X_{2,5} \\ X_{3,3} & X_{3,4} & X_{3,5} \end{bmatrix} \rightarrow \begin{bmatrix} d_{1,[3,4,5]} \\ d_{2,[3,4,5]} \\ d_{3,[3,4,5]} \end{bmatrix} \rightarrow d^*_{[3,4,5]}, \quad \begin{bmatrix} X_{1,4} & X_{1,5} & X_{1,6} \\ X_{2,4} & X_{2,5} & X_{2,6} \\ X_{3,4} & X_{3,5} & X_{3,6} \end{bmatrix} \rightarrow \begin{bmatrix} d_{1,[4,5,6]} \\ d_{2,[4,5,6]} \\ d_{3,[4,5,6]} \end{bmatrix} \rightarrow d^*_{[4,5,6]},$$

$$\begin{bmatrix} X_{1,5} & X_{1,6} & X_{1,7} \\ X_{2,5} & X_{2,6} & X_{2,7} \\ X_{3,5} & X_{3,6} & X_{3,7} \end{bmatrix} \rightarrow \begin{bmatrix} d_{1,[5,6,7]} \\ d_{2,[5,6,7]} \\ d_{3,[5,6,7]} \end{bmatrix} \rightarrow d^*_{[5,6,7]}, \quad \begin{bmatrix} X_{1,6} & X_{1,7} & X_{1,8} \\ X_{2,6} & X_{2,7} & X_{2,8} \\ X_{3,6} & X_{3,7} & X_{3,8} \end{bmatrix} \rightarrow \begin{bmatrix} d_{1,[6,7,8]} \\ d_{2,[6,7,8]} \\ d_{3,[6,7,8]} \end{bmatrix} \rightarrow d^*_{[6,7,8]},$$

where the size of the aggregated details are compared. Suppose that $d^*_{[2,3,4]}$ has the smallest size, then merge the corresponding triplet of columns and update the initial data matrix in (5.10) into:

$$\boldsymbol{S} = \begin{pmatrix} X_{1,1} & s^1_{1,[2,4]} & s^2_{1,[2,4]} & d_{1,[2,3,4]} & X_{1,5} & X_{1,6} & X_{1,7} & X_{1,8} \\ X_{2,1} & s^1_{2,[2,4]} & s^2_{2,[2,4]} & d_{2,[2,3,4]} & X_{2,5} & X_{2,6} & X_{2,7} & X_{2,8} \\ X_{3,1} & s^1_{3,[2,4]} & s^2_{3,[2,4]} & d_{3,[2,3,4]} & X_{3,5} & X_{3,6} & X_{3,7} & X_{3,8} \end{pmatrix}. \tag{5.11}$$

We categorise this transformation into Type 1 (merging three initial smooth coefficient vectors).

Scale $j = 2$. From now on, the "two together" rule is applied. Ignoring any detail coefficient columns in the data matrix in (5.11), the possible triplets of columns for next merging are $(1^{\text{st}}, 2^{\text{nd}}, 3^{\text{rd}})$, $(2^{\text{nd}}, 3^{\text{rd}}, 5^{\text{th}})$, $(5^{\text{th}}, 6^{\text{th}}, 7^{\text{th}})$, $(6^{\text{th}}, 7^{\text{th}}, 8^{\text{th}})$ columns. We note that the triplet of $(3^{\text{rd}}, 5^{\text{th}}, 6^{\text{th}})$ columns cannot be considered as a candidate for next merging under the "two together" rule as this triplet contains only one (not both) of the paired smooth coefficient columns returned by the previous merging. Assume that the triplet of $(5^{\text{th}}, 6^{\text{th}}, 7^{\text{th}})$ columns gives the smallest size of the aggregated detail coefficient $d^*_{[5,6,7]}$ among the four candidates, then we merge them through the orthogonal transformation formulated in (5.22) and now update the data sequence matrix into

$$\boldsymbol{S} = \begin{pmatrix} X_{1,1} & s^1_{1,[2,4]} & s^2_{1,[2,4]} & d_{1,[2,3,4]} & s^1_{1,[5,7]} & s^2_{1,[5,7]} & d_{1,[5,6,7]} & X_{1,8} \\ X_{2,1} & s^1_{2,[2,4]} & s^2_{2,[2,4]} & d_{2,[2,3,4]} & s^1_{2,[5,7]} & s^2_{2,[5,7]} & d_{2,[5,6,7]} & X_{2,8} \\ X_{3,1} & s^1_{3,[2,4]} & s^2_{3,[2,4]} & d_{3,[2,3,4]} & s^1_{3,[5,7]} & s^2_{3,[5,7]} & d_{3,[5,6,7]} & X_{3,8} \end{pmatrix}. \tag{5.12}$$

This transformation is also Type 1.

Scale $j = 3$. We now compare four candidates for merging, the triplet of $(1^{\text{st}}, 2^{\text{nd}}, 3^{\text{rd}})$, $(2^{\text{nd}}, 3^{\text{rd}}, 5^{\text{th}})$, $(3^{\text{rd}}, 5^{\text{th}}, 6^{\text{th}})$ and $(5^{\text{th}}, 6^{\text{th}}, 8^{\text{th}})$ columns of (5.12). To obey the "two together" rule, we should treat two triplets in middle, $(\boldsymbol{s}^1_{\cdot,[2,4]}, \boldsymbol{s}^2_{\cdot,[2,4]}, \boldsymbol{s}^1_{\cdot,[5,7]})$ and $(\boldsymbol{s}^2_{\cdot,[2,4]}, \boldsymbol{s}^1_{\cdot,[5,7]}, \boldsymbol{s}^2_{\cdot,[5,7]})$, together as they contain two sets of paired smooth coefficient columns, $(\boldsymbol{s}^1_{\cdot,[2,4]}, \boldsymbol{s}^2_{\cdot,[2,4]})$ and $(\boldsymbol{s}^1_{\cdot,[5,7]}, \boldsymbol{s}^2_{\cdot,[5,7]})$, where $\boldsymbol{s}^{\cdot}_{\cdot,[p,r]} = (s^{\cdot}_{1,[p,r]}, s^{\cdot}_{2,[p,r]}, \ldots, s^{\cdot}_{n,[p,r]})^\top$. The summary detail coefficient vector for this pair of triplet columns is obtained as

$$\begin{bmatrix} d_{1,[2,4,7]} \\ d_{2,[2,4,7]} \\ d_{3,[2,4,7]} \end{bmatrix} = \begin{bmatrix} \max(|d^1_{1,[2,4,7]}|, |d^2_{1,[2,4,7]}|) \\ \max(|d^1_{2,[2,4,7]}|, |d^2_{2,[2,4,7]}|) \\ \max(|d^1_{3,[2,4,7]}|, |d^2_{3,[2,4,7]}|) \end{bmatrix}.$$

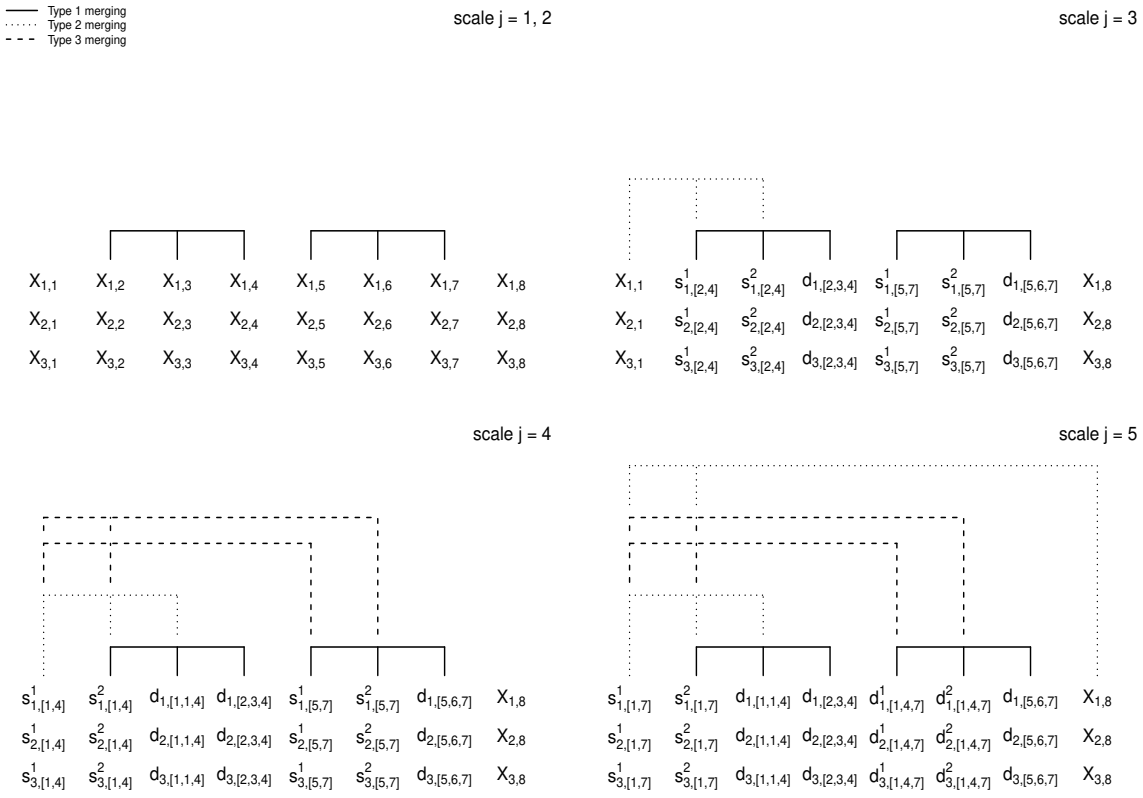


Fig. 5.3 Construction of tree for the example of scenario (S2) in Section 5.2.3; each diagram shows all merges performed up to the given scale.

The corresponding aggregated detail coefficient is obtained as $d^*_{[2,4,7]} = \max_i |d_{i,[2,4,7]}|_{i=1,2,3}$, which is compared with those of other triplets of columns. Now suppose that the triplet of $(1^{\text{st}}, 2^{\text{nd}}, 3^{\text{rd}})$ columns of (5.12) has the smallest size of aggregated details; we merge this triplet of columns and update the data sequence matrix into

$$\boldsymbol{S} = \begin{pmatrix} s^1_{1,[1,4]} & s^2_{1,[1,4]} & d_{1,[1,1,4]} & d_{1,[2,3,4]} & s^1_{1,[5,7]} & s^2_{1,[5,7]} & d_{1,[5,6,7]} & X_{1,8} \\ s^1_{2,[1,4]} & s^2_{2,[1,4]} & d_{2,[1,1,4]} & d_{2,[2,3,4]} & s^1_{2,[5,7]} & s^2_{2,[5,7]} & d_{2,[5,6,7]} & X_{2,8} \\ s^1_{3,[1,4]} & s^2_{3,[1,4]} & d_{3,[1,1,4]} & d_{3,[2,3,4]} & s^1_{3,[5,7]} & s^2_{3,[5,7]} & d_{3,[5,6,7]} & X_{3,8} \end{pmatrix}. \tag{5.13}$$

This transformation is of Type 2.

Scale $j = 4$. We now have two pairs of paired coefficient columns: $(\boldsymbol{s}^1_{\cdot,[1,4]}, \boldsymbol{s}^2_{\cdot,[1,4]})$ and $(\boldsymbol{s}^1_{\cdot,[5,7]}, \boldsymbol{s}^1_{\cdot,[5,7]})$ in (5.13). Therefore, with the "two together" rule in mind, the only possi-

ble options for merging are: to merge the two pairs into $(\boldsymbol{s}^1_{\cdot,[1,4]}, \boldsymbol{s}^2_{\cdot,[1,4]}, \boldsymbol{s}^1_{\cdot,[5,7]}, \boldsymbol{s}^2_{\cdot,[5,7]})$, or to merge $(\boldsymbol{s}^1_{\cdot,[5,7]}, \boldsymbol{s}^1_{\cdot,[5,7]})$ with $\boldsymbol{X}_{\cdot,8}$. Suppose that the first merging is preferred. The merge of $(\boldsymbol{s}^1_{\cdot,[1,4]}, \boldsymbol{s}^2_{\cdot,[1,4]})$ and $(\boldsymbol{s}^1_{\cdot,[5,7]}, \boldsymbol{s}^2_{\cdot,[5,7]})$ into $(\boldsymbol{s}^1_{\cdot,[1,4]}, \boldsymbol{s}^2_{\cdot,[1,4]}, \boldsymbol{s}^1_{\cdot,[5,7]}, \boldsymbol{s}^2_{\cdot,[5,7]})$ is of Type 3 and is performed in two stages as follows. In the first stage, we merge $(\boldsymbol{s}^1_{\cdot,[1,4]}, \boldsymbol{s}^2_{\cdot,[1,4]}, \boldsymbol{s}^1_{\cdot,[5,7]})$ and then update the data matrix temporarily as $\boldsymbol{S} = (\boldsymbol{s}^{1'}_{\cdot,[1,7]}, \boldsymbol{s}^{2'}_{\cdot,[1,7]}, \boldsymbol{d}_{\cdot,[1,1,4]}, \boldsymbol{d}_{\cdot,[2,3,4]}, \boldsymbol{d}^1_{\cdot,[1,4,7]}, \boldsymbol{s}^2_{\cdot,[5,7]}, \boldsymbol{d}_{\cdot,[5,6,7]}, \boldsymbol{X}_{\cdot,8})$. In the second stage, we merge $(\boldsymbol{s}^{1'}_{\cdot,[1,7]}, \boldsymbol{s}^{2'}_{\cdot,[1,7]}, \boldsymbol{s}^2_{\cdot,[5,7]})$, which gives the updated data sequence matrix shown at the bottom right diagram of Figure 5.3. As an aggregated detail coefficient for this merge, we use $d^*_{[1,4,7]} = \max_i |d_{i,[1,4,7]}|_{i=1,2,3}$ where $d_{i,[1,4,7]} = \max(|d^1_{i,[1,4,7]}|, |d^2_{i,[1,4,7]}|)$.

Scale $j = 5$. The only available triplet of columns is now $(\boldsymbol{s}^1_{\cdot,[1,7]}, \boldsymbol{s}^2_{\cdot,[1,7]}, \boldsymbol{X}_{\cdot,8})$, thus we perform this Type 2 merge and update the data sequence matrix into

$$\boldsymbol{S} = \begin{pmatrix} s^1_{1,[1,8]} & s^2_{1,[1,8]} & d_{1,[1,1,4]} & d_{1,[2,3,4]} & d^1_{1,[1,4,7]} & d^2_{1,[1,4,7]} & d_{1,[5,6,7]} & d_{1,[1,7,8]} \\ s^1_{2,[1,8]} & s^2_{2,[1,8]} & d_{2,[1,1,4]} & d_{2,[2,3,4]} & d^1_{2,[1,4,7]} & d^2_{2,[1,4,7]} & d_{2,[5,6,7]} & d_{2,[1,7,8]} \\ s^1_{3,[1,8]} & s^2_{3,[1,8]} & d_{3,[1,1,4]} & d_{3,[2,3,4]} & d^1_{3,[1,4,7]} & d^2_{3,[1,4,7]} & d_{3,[5,6,7]} & d_{3,[1,7,8]} \end{pmatrix}. \quad (5.14)$$

The transformation is completed with the updated data sequence matrix which contains $T - 2 = 6$ columns of detail coefficients and 2 columns of smooth coefficients.

Discussion on the coordinate-wise aggregation of detail coefficients

Before formulating the HiTGUW transform in general, we discuss some properties of the aggregation method used in the transformation. As shown in the two examples above, conceptually, the difference between TrendSegment and HiTS is that we need to aggregate n detail coefficient vectors computed from n univariate data sequences to decide which region should be merged first in the HiTGUW transform. For this purpose, we choose the l_∞ norm and now justify this choice in detail. We recall that a detail coefficient $d_{i,[p,q,r]}$ is computed from a subregion $\{X_{i,p}, \ldots, X_{i,r}\}$ of i^{th} data sequence and

it represents the strength of the corresponding local polynomial trend; the smaller the (absolute) size of the detail, the smaller the local deviation from constancy in scenario (S1) and from linearity in scenario (S2). Therefore, the coordinate-wise pointwise maximum (l_∞-aggregation) of detail coefficient vectors, $d^*_{[p,q,r]} = \max_{i=1,\ldots,n} |d_{i,[p,q,r]}|$, corresponds to a data sequence whose deviation from a local polynomial trend is the strongest among all n univariate data sequences in a fixed subregion $[p, r]$.

As those smooth coefficient vectors corresponding to the smallest "aggregated" detail coefficients have priority in merging in the HiTGUW transform, using l_∞-aggregation allows us to merge the neighbouring regions whose least likely data sequence (i.e. a data sequence corresponding to the maximum size of the detail) is more likely to belong to the same segment in terms of a polynomial trend than the least likely data sequence of other neighbouring regions. In other words, the l_∞-aggregation of the detail coefficients encourages the HiTGUW transform to operate in a way of delaying the merge of regions in which at least one data sequence includes an extremely large size of change. Therefore, as will be supported by our numerical studies in Section 5.4, the HiTS algorithm provides a particularly better performance than other competing methods in the extremely sparse case (i.e. when a very small number of data sequences experiences the changes) in which other competing methods significantly underperform, but become relatively less attractive than other methods when the changes occur in most of the data sequences. We note that our theory in Section 5.3 is not built on a particular assumption on the sparsity level.

Although the l_∞-aggregation enables the HiTS algorithm to capture some sparse changes well, if there exist two types of changes, 1) a sparse but large change and 2) a dense but gentle change, the HiTS algorithm possibly misses a chance of detecting some gentle changes occurred in most of the data sequences. This is because the HiTGUW transform is performed in a way of prioritising the merge of the subregions including a

dense but gentle change and delaying the merge of the subregions containing a sparse but large change, which gives a higher chance to "a sparse but large change" to be survived in thresholding.

In the literature, other possible ways of aggregation have been suggested for CUSUM series e.g. l_2-norm (Horváth and Hušková, 2012) and l_1-norm to the hard-thresholded CUSUM series (Cho and Fryzlewicz, 2015). As opposed to those methods based on the Binary Segmentation (thus they focus on the region corresponding to the "largest" aggregated CUSUM series to operate a divisive algorithm), our agglomerative approach gives priority in merging to the region corresponding to the "smallest" aggregated detail coefficients in which l_∞-norm works well for estimating change-point that is sparse across the panel. Some other existing ways of aggregation or projection for the high-dimensional panel data can be found in Section 2.3.3, however we emphasise that the aggregation of detail-type coefficients for a bottom-up transformation has not previously been studied in the literature.

As is in the TGUW transform introduced in Chapter 4, the HiTGUW transformation also has the "tail-greediness" (Fryzlewicz, 2018b) which allows us to reduce the computational complexity by performing multiple merges over non-overlapping regions in a single pass over the data. More specifically, in scenario (S1), it enables us to perform up to $\lceil \rho\alpha_j \rceil$ merges at each scale j, where α_j is the number of smooth coefficient columns in the data matrix $\boldsymbol{S}$ and $\rho \in (0, 1)$. In scenario (S2), up to $\max\{2, \lceil \rho\alpha_j \rceil\}$ merges are allowed to be performed at each scale j where the lower bound of 2 is essential to permit a Type 3 transformation, which consists of two merges.

In this chapter, a detail coefficient $d^{\cdot}_{i,[p,q,r]}$ will be sometimes referred to as $d^{(j,k)}_i$ or $d^{(j,k)}_{i,[p,q,r]}$, where $j = 1, \ldots, J$ is the scale of the transform at which $d^{\cdot}_{i,[p,q,r]}$ was computed, and $k = 1, \ldots, K(j)$ is the location index of $d^{\cdot}_{i,[p,q,r]}$ within all scale j coefficients. Note

that $d_{i,[p,q,r]}$ is $d_{i,[p,q,r]}$ or $d^1_{i,[p,q,r]}$ or $d^2_{i,[p,q,r]}$ depending on the type of merge in scenario (S2).

Now we are ready to formulate the HiTGUW transformation in general for each scenario.

HiTGUW transformation: general algorithm in scenario (S1)

In general, the HiTGUW algorithm in scenario (S1) is formulated as follows.

1. At each scale j, for each pair of neighbouring smooth coefficients, $(s_{i,[p,q]}, s_{i,[q+1,r]})$, compute the corresponding detail coefficient $d_{i,[p,q,r]}$ for $i = 1, \ldots, n$ and its coordinate-wise aggregation as follows,

$$d_{i,[p,q,r]} = a_{p,q,r}\ s_{i,[p,q]} + b_{p,q,r}\ s_{i,[q+1,r]}, \quad i = 1, \ldots, n, \tag{5.15}$$

$$d^*_{[p,q,r]} = \max_i \left| d_{i,[p,q,r]} \right|_{i=1,\ldots,n}, \tag{5.16}$$

where $p < q < r$. The constants $a_{p,q,r}, b_{p,q,r}$ are the elements of the detail filter $\boldsymbol{h}_{p,q,r} = (a_{p,q,r}, b_{p,q,r})^\top$ where $a_{p,q,r}b_{p,q,r} < 0$. The detail filter should satisfy the condition that the detail coefficient $d_{i,[p,q,r]}$ is zero for any i only if the corresponding raw observations of merged regions, $(X_{i,p}, \ldots, X_{i,r})$, form a constant vector. This implies that the smaller size of detail coefficient we have, the stronger constancy exists in those regions. Another requirement on the detail filter is $a^2_{p,q,r} + b^2_{p,q,r} = 1$ which preserves the orthonormality of the transform. Specifically, those two conditions give the following,

$$a_{p,q,r} = \sqrt{(r-q)/(r-p+1)}, \quad b_{p,q,r} = -\sqrt{(q-p+1)/(r-p+1)}. \tag{5.17}$$

2. Sort the size of the aggregated detail coefficients $d^*_{[p,q,r]}$ obtained in step 1 in non-decreasing order.

3. Extract the (non-aggregated) detail coefficient vector $\left\{d_{i,[p,q,r]}\right\}_{i=1}^{n}$ corresponding to the smallest (aggregated) detail coefficient $d^*_{[p,q,r]}$. We repeat the extraction until $\lceil \rho\alpha_j \rceil$ (or all possible, whichever is the smaller number) detail coefficient vectors have been obtained, as long as the region of the data corresponding to each detail coefficient vector extracted does not overlap with the regions corresponding to the detail coefficient vectors already drawn.
4. For each $d_{i,[p,q,r]}$ extracted in step 3, merge the corresponding smooth coefficients through the orthonormal transform as follows,

$$\begin{pmatrix} s_{i,[p,r]} \\ d_{i,[p,q,r]} \end{pmatrix} = \begin{pmatrix} -b_{p,q,r} & a_{p,q,r} \\ a_{p,q,r} & b_{p,q,r} \end{pmatrix} \begin{pmatrix} s_{i,[p,q]} \\ s_{i,[q+1,r]} \end{pmatrix}, \quad i = 1, \ldots, n. \tag{5.18}$$

5. Go to step 1 and repeat at new scale $j = j + 1$ as long as we have at least two columns of smooth coefficients in the updated data sequence matrix $\boldsymbol{S}$.

HiTGUW transformation: general algorithm in scenario (S2)

In general, the HiTGUW algorithm in scenario (S2) is formulated as follows.

1. At each scale j, find the set of triplet columns of smooth coefficients in $\boldsymbol{S}$ that are candidates for merging under the "two together" rule. Compute the corresponding detail coefficients where it is formulated as follows in any type of merge,

$$d_{i,[p,q,r]} = a_{p,q,r} S^1_{i,p:r} + b_{p,q,r} S^2_{i,p:r} + c_{p,q,r} S^3_{i,p:r}, \quad i = 1, \ldots, n, \tag{5.19}$$

where $p < q < r$, $S^k_{i,p:r}$ is the k^{th} smooth coefficient column of the submatrix $\boldsymbol{S}_{[,p:r]}$ of the dimension $n \times (r - p + 1)$ and the constants $a_{p,q,r}, b_{p,q,r}, c_{p,q,r}$ are the elements of the detail filter $\boldsymbol{h}_{p,q,r} = (a_{p,q,r}, b_{p,q,r}, c_{p,q,r})^\top$. The detail filter produces the weighted sum of a triplet of smooth coefficient columns and should satisfy the condition that

the detail coefficient is zero if and only if the corresponding raw observations over the merged regions have a perfect linear trend. Therefore, the detail coefficient represent the extent of non-linearity in the corresponding region of data which implies that the smaller the size of the detail coefficient, the stronger the linearity of the corresponding data. Specifically, the detail filter $\boldsymbol{h}_{p,q,r} = (a_{p,q,r}, b_{p,q,r}, c_{p,q,r})^\top$ is obtained by solving the following equations,

$$\begin{aligned} a_{p,q,r}\boldsymbol{w}^{c,1}_{p:r} + b_{p,q,r}\boldsymbol{w}^{c,2}_{p:r} + c_{p,q,r}\boldsymbol{w}^{c,3}_{p:r} &= 0, \\ a_{p,q,r}\boldsymbol{w}^{l,1}_{p:r} + b_{p,q,r}\boldsymbol{w}^{l,2}_{p:r} + c_{p,q,r}\boldsymbol{w}^{l,3}_{p:r} &= 0, \\ a^2_{p,q,r} + b^2_{p,q,r} + c^2_{p,q,r} &= 1, \end{aligned} \tag{5.20}$$

where $\boldsymbol{w}^{\cdot,k}_{p:r}$ is k^{th} non-zero element of the subvector $\boldsymbol{w}^{\cdot}_{p:r}$ of length $r-p+1$, and $\boldsymbol{w}^c$ and $\boldsymbol{w}^l$ are weight vectors of constancy and linearity, respectively, whose initial inputs are $\boldsymbol{w}^c_0 = (1, 1, \ldots, 1)^\top$ and $\boldsymbol{w}^l_0 = (1, 2, \ldots, T)^\top$. The last condition in (5.20) preserves the orthonormality of the transform and the detail filter obtained as a solution of (5.20) is unique up to multiplication by -1.

2. Find a summary detail coefficient $d_{i,[p,q,r]} = \max(|d^1_{i,[p,q,r]}|, |d^2_{i,[p,q,r]}|)$ for any pair of detail coefficients constructed by Type 3 merges. Using a summarised sequence of $d_{i,[p,q,r]}$, compute the aggregated detail coefficients,

$$d^*_{[p,q,r]} = \max_i \left| d_{i,[p,q,r]} \right|_{i=1,\ldots,n}. \tag{5.21}$$

3. Sort the size of the aggregated detail coefficients $\left|d^*_{[p,q,r]}\right|$ obtained in step 2 in non-decreasing order.
4. Extract the (non-summarised and non-aggregated) detail coefficients $\left\{\left|d^{\cdot}_{i,[p,q,r]}\right|\right\}^n_{i=1}$ corresponding to the smallest (summarised and aggregated) detail coefficient $\left|d^*_{[p,q,r]}\right|$ where both $\left\{\left|d^1_{i,[p,q,r]}\right|\right\}^n_{i=1}$ and $\left\{\left|d^2_{i,[p,q,r]}\right|\right\}^n_{i=1}$ should be extracted only when $d_{i,[p,q,r]} =$

$\max\left(\left|d^1_{i,[p,q,r]}\right|, \left|d^2_{i,[p,q,r]}\right|\right)$ is extracted. The extraction should be repeated until $\max\{2, \lceil \rho\alpha_j \rceil\}$ (or all possible, whichever is the smaller number) columns of detail coefficients are obtained, as long as the region of the data corresponding to each column of detail coefficients extracted does not overlap with the regions corresponding to the columns of detail coefficients already drawn.

5. For each $\left|d^{\cdot}_{i,[p,q,r]}\right|$ extracted in step 4, merge the corresponding smooth coefficients by updating the corresponding triplet columns in $\boldsymbol{S}$ and the corresponding triplet in $\boldsymbol{w}^c$ and $\boldsymbol{w}^l$ through the orthonormal transforms as follows,

$$\begin{pmatrix} s^1_{i,[p,r]} \\ s^2_{i,[p,r]} \\ d^{\cdot}_{i,[p,q,r]} \end{pmatrix} = \begin{pmatrix} \boldsymbol{\ell}_1^\top \\ \boldsymbol{\ell}_2^\top \\ \boldsymbol{h}^\top \end{pmatrix} \begin{pmatrix} S^1_{i,p:r} \\ S^2_{i,p:r} \\ S^3_{i,p:r} \end{pmatrix} = \Lambda \begin{pmatrix} S^1_{i,p:r} \\ S^2_{i,p:r} \\ S^3_{i,p:r} \end{pmatrix}, \quad i = 1, \ldots, n, \tag{5.22}$$

$$\begin{pmatrix} w^{c,1}_{p,r} \\ w^{c,2}_{p,r} \\ 0 \end{pmatrix} = \Lambda \begin{pmatrix} \boldsymbol{w}^{c,1}_{p:r} \\ \boldsymbol{w}^{c,2}_{p:r} \\ \boldsymbol{w}^{c,3}_{p:r} \end{pmatrix}, \quad \begin{pmatrix} w^{l,1}_{p,r} \\ w^{l,2}_{p,r} \\ 0 \end{pmatrix} = \Lambda \begin{pmatrix} \boldsymbol{w}^{l,1}_{p:r} \\ \boldsymbol{w}^{l,2}_{p:r} \\ \boldsymbol{w}^{l,3}_{p:r} \end{pmatrix}. \tag{5.23}$$

The orthonormal matrix, Λ, of the dimension 3×3 is obtained in the same way as in the TGUW transformation by finding two low-pass filters $(\boldsymbol{\ell}_1^\top, \boldsymbol{\ell}_2^\top)$ which satisfy the orthonormality of Λ.

6. Go to step 1 and repeat at new scale $j = j + 1$ as long as we have at least three columns of smooth coefficients in the updated data sequence matrix $\boldsymbol{S}$.

More in detail, the detail coefficient in (5.19) is formulated for Type 3 merge as follows,

$$\begin{aligned} d^1_{i,[p,q,r]} &= a^1_{p,q,r} s^1_{i,[p,q]} + b^1_{p,q,r} s^2_{i,[p,q]} + c^1_{p,q,r} s^1_{i,[q+1,r]}, \\ d^2_{i,[p,q,r]} &= a^2_{p,q,r} s^{01}_{i,[p,r]} + b^2_{p,q,r} s^{02}_{i,[p,r]} + c^2_{p,q,r} s^2_{i,[q+1,r]}, \quad i = 1, \ldots, n, \end{aligned}$$

where $q > p+1$ and $r > q+2$. As is in the TGUW transform in Chapter 4, after the first detail coefficient vector, $\left\{d^1_{i,[p,q,r]}\right\}_{i=1}^{n}$, is obtained, we instantly update the corresponding triplet columns in $\boldsymbol{S}$ and triplets of $\boldsymbol{w}^c$ and $\boldsymbol{w}^l$ through an orthonormal transform as defined in (5.22) and (5.23). Thus, the second detail filter, $(a^2_{p,q,r}, b^2_{p,q,r}, c^2_{p,q,r})$, is obtained with the updated $\boldsymbol{w}^c$ and $\boldsymbol{w}^l$ in a way that satisfies the conditions in (5.20).

The HiTGUW transform ultimately converts the input data matrix $\boldsymbol{X}$ of the dimension $n \times T$ into the matrix containing one column of smooth coefficients and $T-1$ columns of detail coefficients through $T-1$ orthonormal transforms in scenario (S1), and converts the input data matrix $\boldsymbol{X}$ into the matrix containing 2 columns of smooth coefficients and $T-2$ columns of detail coefficients through $T-2$ orthonormal transforms in scenario (S2). In both scenarios, a detail coefficient $d_i^{(j,k)}$ is the scalar products between $\boldsymbol{X}_i$ and a particularly constructed unbalanced wavelet basis $\psi^{(j,k)}$, where the formal representation is given as $\left\{d_i^{(j,k)} = \langle \boldsymbol{X}_i, \psi^{(j,k)} \rangle_{,i=1,\ldots,n,\ j=1,\ldots,J,\ k=1,\ \ldots,K(j)}\right\}$. The smooth coefficient has a form of $s_{i,[1,T]} = \langle \boldsymbol{X}_i, \psi^{(0,1)} \rangle$ in scenario (S1) and $s^1_{i,[1,T]} = \langle \boldsymbol{X}_i, \psi^{(0,1)} \rangle$, $s^2_{i,[1,T]} = \langle \boldsymbol{X}_i, \psi^{(0,2)} \rangle$ in scenario (S2) for $i = 1, \ldots, n$. The set $\{\psi^{(j,k)}\}$ is an orthonormal unbalanced wavelet basis for $\mathbb{R}^T$.

Computational complexity of HiTGUW

As in the TGUH and the TGUW transforms, the HiTGUW (in both scenarios (S1) and (S2)) includes at most $J = O(\log(T))$ scales as the number of merges depends only on the length of data sequences T, not on the dimension n. The HiTGUW algorithm requires $O(nT)$ operations for computing detail coefficients and aggregating them by finding the coordinate-wise maximum. Sorting the aggregated detail coefficients takes up to $O(T\log(T))$ operations, thus the computational complexity of the HiTGUW transform is obtained as $O(\log(T) \cdot \max(nT, T\log(T)))$, that is equal to $O(nT\log(T))$ if $n > \log(T)$ and is same as that of TGUW, $O(T\log^2(T))$, if $n < \log(T)$.

5.2.4 Thresholding

Through the thresholding, we wish to estimate the underlying signal $\{\boldsymbol{f}_i\}_{i=1}^n$ in (5.1) by estimating $\mu_i^{(j,k)} = \langle \boldsymbol{f}_i, \psi^{(j,k)} \rangle$ for $i = 1, \ldots, n$ where $\psi^{(j,k)}$ is an orthonormal unbalanced wavelet basis constructed in the HiTGUW transform from the data. In both scenarios (S1) and (S2), the HiTGUW detail coefficients are thresholded under the "connected" rule which prunes the branches of the (aggregated) HiTGUW detail coefficients if and only if the (aggregated) detail coefficient itself and all of its (aggregated) children coefficients fall below a certain threshold in absolute value. Pruning the branch of the aggregated detail coefficients implies that all elements of the corresponding (non-aggregated) detail coefficient vector are set to zero. After the "connected" rule is applied, only in scenario (S2), we use the "two together" rule that is similar to the one in Section 4.2.3 except for the fact that it targets paired vectors of detail coefficients rather than pairs of detail coefficients. The "two together" rule means that both such detail coefficient vectors should be kept if at least one (aggregated) detail coefficient survives the initial hard thresholding. This is a necessary condition as a pair of Type 3 detail coefficient vectors corresponds to a single merge of two adjacent regions.

The "connected" rule for scenarios (S1) and (S2)

Throughout the thresholding procedure under the "connected" rule the estimator $\mu_i^{(j,k)}$ is obtained as

$$\hat{\mu}_i^{(j,k)} = d_{i,[p,q,r]}^{(j,k)} \cdot \mathbb{I}\left\{ \exists (j',k') \in \mathcal{C}_{j,k} \quad \left| d_{[p',q',r']}^{*(j',k')} \right| > \lambda \right\}, \quad i = 1, \ldots, n, \tag{5.24}$$

where $\mathbb{I}$ is an indicator function and

$$\mathcal{C}_{j,k} = \Big\{(j',k'), j' = 1,\dots,j, k' = 1,\dots,K(j') : d^{*(j',k')}_{[p',q',r']} \text{ is such that } [p',r'] \subseteq [p,r]\Big\}. \tag{5.25}$$

The "two together" rule only for scenario (S2)

Only in scenario (S2), let the estimators $\left\{\hat{\mu}_i^{(j,k)}\right\}_{i=1}^n$ in (5.24) be the initial estimators $\left\{\hat{\mu}_{i,0}^{(j,k)}\right\}_{i=1}^n$ and apply the "two together" rule to obtain the final estimators $\left\{\hat{\mu}_i^{(j,k)}\right\}_{i=1}^n$. We note that two detail coefficients, $d^{(j,k)}_{i,[p,q,r]}$ and $d^{(j',k+1)}_{i',[p',q',r']}$ are called "paired" when they are formed by Type 3 merges and when $(i,j,p,q,r) = (i',j',p',q',r')$. For $i = 1,\dots,n$, the "two together" rule is formulated as below,

$$\hat{\mu}_i^{(j,k)} = \begin{cases} \hat{\mu}_{i,0}^{(j,k)}, & \text{if } d^{(j,k)}_{i,[p,q,r]} \text{ is not paired,} \\ \hat{\mu}_{i,0}^{(j,k)}, & \text{if } d^{(j,k)}_{i,[p,q,r]} \text{ is paired with } d^{(j,k')}_{i,[p,q,r]} \text{ and both } \hat{\mu}_{i,0}^{(j,k)} \text{ and } \hat{\mu}_{i,0}^{(j,k')} \text{ are} \\ & \text{zero or non-zero,} \\ d_i^{(j,k)}, & \text{if } d^{(j,k)}_{i,[p,q,r]} \text{ is paired with } d^{(j,k')}_{i,[p,q,r]} \text{ and } \hat{\mu}_{i,0}^{(j,k')} \neq 0 \text{ and } \hat{\mu}_{i,0}^{(j,k)} = 0. \end{cases} \tag{5.26}$$

The application of the "connected" rule in scenario (S1) ensures that $\tilde{\boldsymbol{f}}_i$ is a piecewise-constant function composed of sample means for each estimated segment for all $i = 1,\dots,n$. Similarly, the usage of both "connected" and "two together" rules in scenario (S2) guarantees that $\tilde{\boldsymbol{f}}_i$ is a piecewise-linear function composed of best linear fits (in the least-squares sense) for each estimated interval of linearity for all $i = 1,\dots,n$.

5.2.5 Inverse HiTGUW transformation

The estimator $\{\tilde{\boldsymbol{f}}_i\}_{i=1}^n$ of the true signal $\{\boldsymbol{f}_i\}_{i=1}^n$ in (5.1) is obtained by inverting (= transposing) the orthonormal transformations, (5.18) in scenario (S1) and (5.22) in

scenario (S2), in reverse order to that in which they were originally performed. This inverse HiTGUW transformation is referred to as HiTGUW^{-1}, and formulated for each scenario as follows,

Scenario (S1)

$$\tilde{\boldsymbol{f}}_i = \text{HiTGUW}^{-1}\Big\{ \hat{\mu}_i^{(j,k)}, j = 1, \ldots, J,\ k = 1, \ldots, K(j) \parallel s_{i,[1,T]} \Big\},\ i = 1, \ldots, n, \tag{5.27}$$

Scenario (S2)

$$\tilde{\boldsymbol{f}}_i = \text{HiTGUW}^{-1}\Big\{ \hat{\mu}_i^{(j,k)}, j = 1, \ldots, J,\ k = 1, \ldots, K(j) \parallel s^1_{i,[1,T]}, s^2_{i,[1,T]} \Big\},\ i = 1, \ldots, n, \tag{5.28}$$

where $\hat{\mu}_i^{(j,k)}$ is in (5.24) in scenario (S1) and (5.26) in scenario (S2), and $\parallel$ denotes vector concatenation.

5.2.6 Post-processing for consistent estimation

As will be specified in Theorems 5.1 and 5.2 in Section 5.3, the piecewise-constant estimator in (5.27) and the piecewise-linear one in (5.28) possibly overestimate the number of change-points. To get rid of the spurious estimated change-points and to achieve the consistency of the number and locations of the estimated change-points, we propose the modified post-processing framework of Fryzlewicz (2018b) in scenario (S1) and that of TrendSegment in Chapter 4 in scenario (S2). The post-processing methodology contains two stages and both scenarios (S1) and (S2) are considered in each stage.

Stage 1

In this stage, we execute three steps, HiTGUW transform, thresholding and inverse HiTGUW transform, again to the estimated function $\tilde{\boldsymbol{f}}$ in (5.27) or (5.28) (depending on the scenario). Using $\tilde{\boldsymbol{f}}$ as an input data matrix, the HiTGUW transform is performed as presented in Section 5.2.3, but in a greedy rather than tail-greedy way such that only one detail coefficient vector $\left\{d_i^{(j,1)}\right\}_{i=1}^n$ is produced at each scale j, and thus $K(j) = 1$ for all j. We continue to produce detail coefficient until the first (aggregated) detail coefficient such that $|d^{*(j,1)}| > \lambda$ is attained and once that condition is satisfied, stop merging and relabel the surviving change-points as $(\tilde{\tilde{\eta}}_1, \tilde{\tilde{\eta}}_2, \ldots, \tilde{\tilde{\eta}}_{\tilde{\tilde{N}}})$. The new estimator is referred to as $\tilde{\tilde{\boldsymbol{f}}}$ and we note that λ is the parameter used in thresholding in Section 5.2.4. For each scenario, the new estimator $\left\{\tilde{\tilde{\boldsymbol{f}}}_i\right\}_{i=1}^n$ is constructed as follows,

Scenario (S1):

$$\tilde{\tilde{f}}_{i,t} = \frac{1}{\tilde{\tilde{\eta}}_\ell - \tilde{\tilde{\eta}}_{\ell-1}} \sum_{t=\tilde{\tilde{\eta}}_{\ell-1}+1}^{\tilde{\tilde{\eta}}_\ell} X_{i,t} \quad \text{for} \quad t \in \left[\tilde{\tilde{\eta}}_{\ell-1}+1, \tilde{\tilde{\eta}}_\ell\right], \quad i = 1, \ldots, n, \ \ell = 1, \ldots, \tilde{\tilde{N}}, \tag{5.29}$$

Scenario (S2):

$$\tilde{\tilde{f}}_{i,t} = \tilde{\tilde{\theta}}^1_{i,\ell} + \tilde{\tilde{\theta}}^2_{i,\ell}\, t \quad \text{for} \quad t \in \left[\tilde{\tilde{\eta}}_{\ell-1}+1, \tilde{\tilde{\eta}}_\ell\right], \quad i = 1, \ldots, n, \ \ell = 1, \ldots, \tilde{\tilde{N}}, \tag{5.30}$$

where $\tilde{\tilde{\eta}}_0 = 0$, $\tilde{\tilde{\eta}}_{\tilde{\tilde{N}}+1} = T$ and $(\tilde{\tilde{\theta}}^1_{i,\ell}, \tilde{\tilde{\theta}}^2_{i,\ell})$ are the OLS estimators of the corresponding pairs $\left\{(t, X_{i,t}),\ t \in [\tilde{\tilde{\eta}}_{i-1}+1, \tilde{\tilde{\eta}}_i]\right\}$. In both scenarios, when the region under consideration only contains a single data point $X_{\cdot,t_0}$, we simply set $\tilde{\tilde{f}}_{i,t_0} = X_{i,t_0}$.

Stage 2

In the second stage, we examine the regions containing only one estimated change-point to check for its significance. We transform the estimator $\tilde{\tilde{\boldsymbol{f}}}$ obtained in Stage 1 with

change-points $(\tilde{\tilde{\eta}}_1, \tilde{\tilde{\eta}}_2, \ldots, \tilde{\tilde{\eta}}_{\tilde{\tilde{N}}})$ into the final estimator $\hat{\boldsymbol{f}}$ with corresponding change-points $(\hat{\eta}_1, \hat{\eta}_2, \ldots, \hat{\eta}_{\hat{N}})$ by pruning. For each $\ell = 1, \ldots, \tilde{\tilde{N}}$, compute the aggregated detail coefficient $d^*_{[p_\ell, q_\ell, r_\ell]}$ as described in (5.16) and (5.21) for scenarios (S1) and (S2), respectively, where $p_\ell = \left\lfloor \frac{\tilde{\tilde{\eta}}_{\ell-1} + \tilde{\tilde{\eta}}_\ell}{2} \right\rfloor + 1$, $q_i = \tilde{\tilde{\eta}}_\ell$ and $r_\ell = \left\lceil \frac{\tilde{\tilde{\eta}}_\ell + \tilde{\tilde{\eta}}_{\ell+1}}{2} \right\rceil$. Find the minimiser $\ell_0 = \arg\min_\ell \left| d^*_{[p_\ell, q_\ell, r_\ell]} \right|$ and if the condition, $\left| d^*_{[p_{\ell_0}, q_{\ell_0}, r_{\ell_0}]} \right| \leq \lambda$, is satisfied then remove $\tilde{\tilde{\eta}}_{\ell_0}$ and set $\tilde{\tilde{N}} := \tilde{\tilde{N}} - 1$ where λ is as in Stage 1. After removing one change-point, relabel the remaining ones with the subscripts $\ell = 1, \ldots, \tilde{\tilde{N}}$ under the convention $\tilde{\tilde{\eta}}_0 = 0$, $\tilde{\tilde{\eta}}_{\tilde{\tilde{N}}+1} = T$. We repeat to prune while we can find ℓ_0 which satisfies the condition $\left| d^*_{[p_{\ell_0}, q_{\ell_0}, r_{\ell_0}]} \right| \leq \lambda$, otherwise, stop and set $\hat{N}$ as the number of detected change-points. Relabel the change-points $\hat{\eta}_i$ in increasing order for $i = 0, \ldots, \hat{N}+1$ with the convention $\hat{\eta}_0 = 0$ and $\hat{\eta}_{\hat{N}+1} = T$. For each scenario, the final estimator is constructed as follows,

Scenario (S1):

$$\hat{f}_{i,t} = \frac{1}{\hat{\eta}_\ell - \hat{\eta}_{\ell-1}} \sum_{t=\hat{\eta}_{\ell-1}+1}^{\hat{\eta}_\ell} X_{i,t} \quad \text{for} \quad t \in \left[\hat{\eta}_{\ell-1} + 1, \hat{\eta}_\ell\right], \quad i = 1, \ldots, n, \ \ell = 1, \ldots, \hat{N}, \tag{5.31}$$

Scenario (S2):

$$\hat{f}_{i,t} = \hat{\theta}^1_{i,\ell} + \hat{\theta}^2_{i,\ell}\, t \quad \text{for} \quad t \in \left[\hat{\eta}_{\ell-1} + 1, \hat{\eta}_\ell\right], \quad i = 1, \ldots, n, \ \ell = 1, \ldots, \hat{N}, \tag{5.32}$$

where $\hat{\eta}_0 = 0$, $\hat{\eta}_{\hat{N}+1} = T$ and $(\hat{\theta}^1_{i,\ell}, \hat{\theta}^2_{i,\ell})$ are the OLS estimators of the corresponding pairs $\{(t, X_{i,t}),\ t \in [\hat{\eta}_{i-1} + 1, \hat{\eta}_i]\}$, with the exception for point anomalies as described in Stage 1 above. Through these two stages of post-processing, the consistency of the number and the locations of the estimated change-points is achieved, and further details can be found in Section 5.3.

In Sections 5.4 and 5.5, we disable Stages 1 and 2 of post-processing in scenario (S1) and disable only Stage 2 of post-processing in scenario (S2) by default. From our empirical experiences, Stage 1 rarely makes a difference in practice in scenario

(S1) but is useful for removing the overestimated change-points in scenario (S2) with an additional computational cost. In both scenarios, Stage 2 tends to over-prune change-point estimates.

5.3 Theoretical results

In this section, we study the l_2 consistency of three estimators, $\left\{\tilde{\boldsymbol{f}}_i\right\}_{i=1}^n$, $\left\{\tilde{\tilde{\boldsymbol{f}}}_i\right\}_{i=1}^n$ and $\left\{\hat{\boldsymbol{f}}_i\right\}_{i=1}^n$, obtained in Section 5.2 where the l_2 risk of any set of estimators $\left\{\tilde{\boldsymbol{f}}_i\right\}_{i=1}^n$ is defined as $\left\|\tilde{\boldsymbol{f}} - \boldsymbol{f}\right\|_{n,T}^2 = \frac{1}{n}\frac{1}{T}\sum_{i=1}^n\sum_{t=1}^T\left(\tilde{f}_{i,t} - f_{i,t}\right)^2$ and $\boldsymbol{f}$ is the underlying signal in (5.1).

5.3.1 Under temporal independence and general cross-sectional dependence

We first examine the case when the Gaussian random errors in model (5.1) have temporal independence but possibly have cross-sectional dependence. The nature of the cross-sectional dependence is general in that no specific structure is given and the theoretical results stated in this section do not necessarily need the assumption of cross-sectional independence. We first make the following assumptions.

Assumption 5.1 *Var*$(\boldsymbol{\varepsilon}_i) = \sigma_i^2 = 1$ *for all* i *in model* (5.1).

Assumption 5.2 *Let the threshold take the form of* $\lambda = C_1\{2\log(nT)\}^{1/2}$ *with a constant* C_1 *large enough.*

Assumption 5.3 *The dimensionality* n *satisfies* $n \sim T^\alpha$ *for some fixed* $\alpha \in (0, \infty)$.

As is in Assumption 4.1, we assume that $\mathrm{Var}(\boldsymbol{\varepsilon}_i)$ is known. This is because if it is unknown in practice it can usually be estimated using Median Absolute Deviation

(MAD) (Hampel, 1974). Even if σ_i^2s vary across data sequences X_i, we can simply normalise each data sequence by its MAD estimator and more details can be found in Section 5.4.1. It is reasonable to consider the case that the errors have the temporal dependence or a specific form of cross-sectional dependence, thus in Sections 5.3.2 and 5.3.3, we explore how our method works when those relaxations are given. The threshold in Assumption 5.2 has the generalised form ($\lambda = C_1\sigma\{2\log(nT)\}^{1/2}$) and is similar to that in Assumption 4.2 in Section 4.3 except the fact that $\log(T)$ is replaced with $\log(nT)$. This is related to the number of Gaussian components considered, which is T for a univariate time series in Chapter 4 but nT for a high-dimensional panel data that increases with both the length T and the dimension n of the data. The optimal value of the constant C_1 for the practical application of TrendSegment procedure will be specified in Section 5.4.1. In Assumption 5.3, the dimensionality n can increase with T at a polynomial rate, which is necessary for the significant detail coefficients (i.e. those corresponding to the true change-points) to be survived from the thresholding stage. For example, if n can increase with T at an exponential rate (i.e. $n \sim e^T$), then the threshold will become too large, thus those significant details coefficients may be annihilated in thresholding.

We investigate the l_2 behaviour of $\left\{\tilde{\boldsymbol{f}}_i\right\}_{i=1}^n$ in (5.27) and (5.28) returned by the inverse HiTGUW transformation.

Theorem 5.1 *$\{\boldsymbol{X}_i\}_{i=1}^n$ follows model (5.1) in scenario (S1) and $\{\tilde{\boldsymbol{f}}_i\}_{i=1}^n$ is the estimator in (5.27). Then under Assumptions 5.1-5.3, we have*

$$\|\tilde{\boldsymbol{f}} - \boldsymbol{f}\|_{n,T}^2 \;\leq\; C_1^2\,\frac{1}{T}\,\log(nT)\left\{2 + 8N\,\lceil\,\log(T)/\log(1-\rho)^{-1}\,\rceil\right\}, \tag{5.33}$$

with probability approaching to 1 as $n, T \to \infty$ and the piecewise-constant estimator $\{\tilde{\boldsymbol{f}}_i\}_{i=1}^n$ in (5.27) contains $\tilde{N} \leq CN\log(T)$ change-points where C is a constant.

Theorem 5.2 *$\{\boldsymbol{X}_i\}_{i=1}^n$ follows model (5.1) in scenario (S2) and $\{\tilde{\boldsymbol{f}}_i\}_{i=1}^n$ is the estimator in (5.28). Then under Assumptions 5.1-5.3, we have*

$$\|\tilde{\boldsymbol{f}}-\boldsymbol{f}\|_{n,T}^2 \;\leq\; C_1^2\,\frac{1}{T}\,\log(nT)\left\{4+8N\left\lceil \log(T)/\log\left((1-\rho)^{-1}\right)+\log(2)/\log(1-\rho)\right\rceil\right\}, \tag{5.34}$$

with probability approaching to 1 as $n,T\to\infty$ and the piecewise-linear estimator $\{\tilde{\boldsymbol{f}}_i\}_{i=1}^n$ in (5.28) contains $\tilde{N}\leq CN\log(T)$ change-points where C is a constant.

In both scenarios (S1) and (S2), $\tilde{\boldsymbol{f}}$ is l_2 consistent if $N=O(1)$. The l_2 consistency shown in Theorems 5.1 and 5.2 is guaranteed by the "tail-greediness" of the HiTGUW transform. In other words, if we merge only one pair (in scenario (S1)) or one triplet (in scenario (S2)) at each scale, then the consistency is not achieved.

Now we move onto the estimators $\tilde{\tilde{\boldsymbol{f}}}$ in (5.29) and (5.30) obtained in the first stage of post-processing.

Theorem 5.3 *$\{\boldsymbol{X}_i\}_{i=1}^n$ follows model (5.1) in either scenario (S1) or scenario (S2) and $\left\{\tilde{\tilde{\boldsymbol{f}}}_i\right\}_{i=1}^n$ is the estimator either in (5.29) in scenario (S1) or (5.30) in scenario (S2). Then under Assumptions 5.1-5.3, we have $\left\|\tilde{\tilde{\boldsymbol{f}}}-\boldsymbol{f}\right\|_{n,T}^2 = O\left(NT^{-1}\log(T)\log(nT)\right)$ with probability approaching to 1 as $n,T\to\infty$ and there exist at most two change-points between true change-points, $(\eta_\ell,\eta_{\ell+1})$ for $\ell=0,\ldots,N$, which satisfies $\tilde{\tilde{N}}\leq 2(N+1)$ where $\eta_0=0$ and $\eta_{N+1}=T$.*

We see that $\tilde{\tilde{\boldsymbol{f}}}$ is l_2 consistent, but inconsistent for the number of change-points. The consistency of the estimated number and locations of the change-points is established under the following conditions.

Assumption 5.4 *The number of true change-points, N, is finite.*

Assumption 5.5 *Consider the scenario (S1) when $f_{i,t}$ is in (5.3). Let $\Delta_{n,T}=\min_\ell\left\{\left(\underline{f}_{n,T}^\ell\right)^2\cdot\delta_{n,T}^\ell\right\}$ where $\underline{f}_{n,T}^\ell=\min_{i:i\in\Omega_\ell}\left\{\min\left(|f_{i,\eta_{\ell+1}}-f_{i,\eta_\ell}|,|f_{i,\eta_\ell}-f_{i,\eta_{\ell-1}}|\right)\right\}$*

and $\delta_{n,T}^{\ell} = \min\left(|\eta_{\ell} - \eta_{\ell-1}|, |\eta_{\ell+1} - \eta_{\ell}|\right)$. *Assume that* $nTR_{n,T} = o\left(\Delta_{n,T}\right)$ *where* $\left\|\tilde{\tilde{\boldsymbol{f}}} - \boldsymbol{f}\right\|_{n,T}^2 = O_p(R_{n,T})$ *is as in Theorem 5.3 and* $\tilde{\tilde{\boldsymbol{f}}}$ *is the estimator in* (5.29).

Assumption 5.6 *Consider the scenario (S2) when* $f_{i,t}$ *is in* (5.4). *Let* $\Delta_{n,T} = \min_{\ell}\left\{\left(\underline{f}_{n,T}^{\ell}\right)^{2/3} \cdot \delta_{n,T}^{\ell}\right\}$ *where* $\underline{f}_{n,T}^{\ell} = \min_{i:i\in\Omega_{\ell}}\left\{\min\left(|f_{i,\eta_{\ell+1}} - 2f_{i,\eta_{\ell}} + f_{i,\eta_{\ell-1}}|, |f_{i,\eta_{\ell+2}} - 2f_{i,\eta_{\ell+1}} + f_{i,\eta_{\ell}}|\right)\right\}$ *and* $\delta_{n,T}^{\ell} = \min\left(|\eta_{\ell} - \eta_{\ell-1}|, |\eta_{\ell+1} - \eta_{\ell}|\right)$. *Assume that* $n^{1/3}T^{1/3}R_{n,T}^{1/3} = o\left(\Delta_{n,T}\right)$ *where* $\left\|\tilde{\tilde{\boldsymbol{f}}} - \boldsymbol{f}\right\|_{n,T}^2 = O_p(R_{n,T})$ *is as in Theorem 5.3 and* $\tilde{\tilde{\boldsymbol{f}}}$ *is the estimator in* (5.30).

Similar to Assumption 4.3, we control the number of true change-points to be finite in Assumption 5.4. Assumptions 5.5 and 5.6 quantify the difficulty of detecting a change-point in terms of distance from its neighbouring change-points and size of the change in a similar way that Assumption 4.4 does for a univariate data sequence, but the relevant conditions in Assumptions 5.5 and 5.6 are imposed by aggregating the quantified difficulties across the panel.

We finally describe the final estimators $\hat{\boldsymbol{f}}$ in (5.31) and (5.32).

Theorem 5.4 $\{\boldsymbol{X}_i\}_{i=1}^n$ *follows model* (5.1) *in scenario (S1) and* $\left(\{\hat{\boldsymbol{f}}_i\}_{i=1}^n, \hat{N}\right)$ *are the estimators in* (5.31). *Then under Assumptions 5.1-5.5, we have*

$$\mathbb{P}\left(\hat{N} = N, \quad \max_{\ell=1,\ldots,N}\left\{|\hat{\eta}_{\ell} - \eta_{\ell}| \cdot \left(\underline{f}_{n,T}^{\ell}\right)^2\right\} \leq CnTR_{n,T}\right) \to 1, \tag{5.35}$$

as $n, T \to \infty$ *where* C *is a constant.*

Theorem 5.5 $\{\boldsymbol{X}_i\}_{i=1}^n$ *follows model* (5.1) *in scenario (S2) and* $\left(\{\hat{\boldsymbol{f}}_i\}_{i=1}^n, \hat{N}\right)$ *are the estimators in* (5.32). *Then under Assumptions 5.1-5.4 and 5.6, we have*

$$\mathbb{P}\left(\hat{N} = N, \quad \max_{\ell=1,\ldots,N}\left\{|\hat{\eta}_{\ell} - \eta_{\ell}| \cdot \left(\underline{f}_{n,T}^{\ell}\right)^{2/3}\right\} \leq Cn^{1/3}T^{1/3}R_{n,T}^{1/3}\right) \to 1, \tag{5.36}$$

as $n, T \to \infty$ *where* C *is a constant.*

Theorems 5.4 and 5.5 indicate that when point anomalies exist in the set of true change-points under any scenario, a point anomaly η_k and its neighbouring change-point $\eta_{k-1} = \eta_k - 1$ can be detected exactly at their true locations only if the corresponding $\underline{f}^{\ell}_{n,T}$s satisfy the condition $\min\left(\underline{f}^{k}_{n,T}, \underline{f}^{k-1}_{n,T}\right) \gtrsim \sqrt{n\log(T)\log(nT)}$.

Regarding how much the results of this section depend on the Gaussian assumption, we first emphasise that the size of the threshold λ in Assumption 5.2 is closely associated with the tail bound for the standard normal distribution. As this threshold plays an important role in having the bound $NT^{-1}\log(T)\log(nT)$ in Theorems 5.1 and 5.2 which affects the results of the following Theorems 5.3-5.5, the extension to the non-Gaussian distributions can be considered as long as we obtain an appropriate threshold from its tail bound and the corresponding threshold achieves the l_2 consistency results shown in Theorems 5.1 and 5.2.

5.3.2 Under a specific form of temporal dependence and general cross-sectional dependence

In this section, we extend our method to a more realistic setting when the noise is dependent across the time. We consider the case where the errors of the i^{th} data sequence, $\boldsymbol{\varepsilon}_i$ in model (5.1), form a stationary Gaussian process with the autocorrelation function $\rho_i(k)$ for $i = 1, \ldots, n$, where the nature of the cross-sectional dependence is general. We first make the following assumptions.

Assumption 5.7 *For each i, $\boldsymbol{\varepsilon}_i$ in model (5.1) denotes a stationary Gaussian process with the autocorrelation functions $\rho_i(k)$, satisfying $R = \max_i \sum_{k=-\infty}^{\infty} |\rho_i(k)| < \infty$.*

Assumption 5.8 *Let the threshold take the form of $\lambda = C_3\{2R\log(nT)\}^{1/2}$ with a constant C_3 large enough, where R is as in Assumption 5.7.*

Corollary 5.1 *Suppose* $\{\boldsymbol{X}_i\}_{i=1}^n$ *follows model* (5.1) *in scenario (S1), then under Assumptions 5.1, 5.3-5.5, 5.7-5.8, the conclusions of Theorems 5.1, 5.3 and 5.4 still hold with different constants.*

Corollary 5.2 *Suppose* $\{\boldsymbol{X}_i\}_{i=1}^n$ *follows model* (5.1) *in scenario (S2), then under Assumptions 5.1, 5.3-5.4, 5.6, 5.7-5.8, the conclusions of Theorems 5.2, 5.3 and 5.5 still hold with different constants.*

Corollaries 5.1-5.2 imply that the conclusions of Theorems 5.2, 5.3 and 5.5 still holds with the threshold given in Assumption 5.8, therefore those theorems constructed under the temporal independence assumption of the noise are special cases of those obtained under the dependent noise setting formulated in Assumption 5.7, because we have $R = \max_i \sum_{k=-\infty}^{\infty} |\rho_i(k)| = 1$ when the errors of any time series component are independent. The proofs of Corollaries 5.1-5.2 can be found in Section 5.6.

5.3.3 Under temporal independence and a specific form of cross-sectional dependence

We now assume that the errors are dependent across the panel and the noise $\boldsymbol{\varepsilon}_1', \boldsymbol{\varepsilon}_2', \ldots, \boldsymbol{\varepsilon}_T'$ are n-dimensional random vectors sampled from $\boldsymbol{\varepsilon}_t' \sim N_n(\mathbf{0}, \Sigma)$ where Σ is a positive matrix of the dimension $n \times n$ in which all the elements are strictly positive. Since we only consider the dependence across the panel, the errors within any data sequence are assumed to be independent. As shown in the following Corollaries 5.3 and 5.4, HiTS keeps its consistency in estimating the number and the locations of change-points if a specific structure of the cross-sectional dependence is assumed. The proofs of Corollaries 5.3 and 5.4 can be found in Section 5.6.

Assumption 5.9 *In model (5.1), the noise vectors $\boldsymbol{\varepsilon}'_1, \boldsymbol{\varepsilon}'_2, \ldots, \boldsymbol{\varepsilon}'_T$ are n-dimensional random vectors sampled from $\boldsymbol{\varepsilon}'_t \sim N_n(\mathbf{0}, \Sigma)$ where Σ is a positive matrix of the dimension of $n \times n$ in which all the elements are strictly positive.*

Corollary 5.3 *Suppose $\{\boldsymbol{X}_i\}_{i=1}^n$ follows model (5.1) in scenario (S1), then under Assumptions 5.1-5.5, 5.9, the conclusions of Theorems 5.1, 5.3 and 5.4 still hold with different constants.*

Corollary 5.4 *Suppose $\{\boldsymbol{X}_i\}_{i=1}^n$ follows model (5.1) in scenario (S2), then under Assumptions 5.1-5.4, 5.6, 5.9, the conclusions of Theorems 5.2, 5.3 and 5.5 still hold with different constants.*

5.4 Simulations

Although the asymptotic behaviours of the HiTS algorithm are studied under various dependence structures in Section 5.3, in simulations, we only consider the case when the noise $\boldsymbol{\varepsilon}'_1, \boldsymbol{\varepsilon}'_2, \ldots, \boldsymbol{\varepsilon}'_T$ are n-dimensional random vectors sampled from $\boldsymbol{\varepsilon}'_t \sim N_n(\mathbf{0}, \mathbf{I}_n)$, which can be classified into the case stated in Section 5.3.1.

5.4.1 Parameter choice

Choice of threshold λ. As stated in Theorems 5.1 and 5.2, we use the threshold of the form $\lambda = C\sigma\sqrt{2\log(nT)}$. In the implementation of the HiTS procedure, we assume that σ is unknown and can vary across data sequences X_i. We first estimate each σ_i using the Median Absolute Deviation (MAD) estimator (Hampel, 1974) defined as $\hat{\sigma}_i = \text{Median}(|X_{i,2} - X_{i,1}|, \ldots, |X_{i,T} - X_{i,T-1}|)/(\Phi^{-1}(3/4)\sqrt{2})$ in scenario (S1) and $\hat{\sigma}_i = \text{Median}(|X_{i,1} - 2X_{i,2} + X_{i,3}|, \ldots, |X_{i,T-2} - 2X_{i,T-1} + X_{i,T}|)/(\Phi^{-1}(3/4)\sqrt{6})$ in scenario (S2) for $i = 1, \ldots, n$ where Φ^{-1} is the quantile function of the Gaussian distribution. Then we normalise each data sequence by its estimated standard deviation

and use the threshold $\lambda = C\sqrt{2\log(nT)}$ by replacing σ to 1. We use $C = 1.2$ as a default in both scenarios as it empirically led to the best performance over the range $C \in [1, 1.4]$.

Choice of the "tail-greediness" parameter. $\rho \in (0, 1)$ is the parameter which decides the number of merges performed in a single pass over the data. We use $\rho = 0.04$ as a default in the simulation study and data analyses as our empirical experience shows that the best performance is achieved in the range $\rho \in (0, 0.05]$.

Level of cross-sectional sparsity. In both scenarios (S1) and (S2), we consider the simulation settings, $n = (100, 300, 500)$ and sparsity=$(0.01, 0.1, 0.7)$, where $\lfloor \text{sparsity} \cdot n \rceil = |\cup_{\ell=1}^{N} \Omega_\ell|$ is the number of coordinates which experience the changes, Ω_ℓ is defined in (5.3) in scenario (S1) and (5.4) in scenario (S2) and $\lfloor x \rceil$ is the nearest integer from x. Under the size of n, the simulation setting can be high-dimensional depending on the length of data sequences T, which varies with signal considered in Section 5.4.2.

Level of overlap between coordinates. Similar to those done in Wang and Samworth (2018), in what follows, we consider three different levels of overlap between the coordinates, (1) "complete-overlap" case in which all true change-points, $\eta_1, \ldots, \eta_N$, occur in all coordinates those including change-points, $\boldsymbol{f}_1, \ldots, \boldsymbol{f}_{|\cup_{\ell=1}^{N}\Omega_\ell|}$, (2) "half-overlap" case in which the first half of true change-points, $\eta_1, \ldots, \eta_{\lfloor N/2 \rceil}$, occurs in all coordinates those including change-points, $\boldsymbol{f}_1, \ldots, \boldsymbol{f}_{|\cup_{\ell=1}^{N}\Omega_\ell|}$, while the last half of true change-points, $\eta_{\lfloor N/2 \rceil+1}, \ldots, \eta_N$, occurs only in the half of those coordinates having change-points, $\boldsymbol{f}_1, \ldots, \boldsymbol{f}_{\lfloor |\cup_{\ell=1}^{N}\Omega_\ell|/2 \rceil}$, and (3) "no-overlap" case in which the first half of true change-points, $\eta_1, \ldots, \eta_{\lfloor N/2 \rceil}$, occurs only in the first half of those coordinates including change-points, $\boldsymbol{f}_1, \ldots, \boldsymbol{f}_{\lfloor |\cup_{\ell=1}^{N}\Omega_\ell|/2 \rceil}$, and the last half of true change-points, $\eta_{\lfloor N/2 \rceil+1}, \ldots, \eta_N$, occurs only in the last half of those coordinates including change-points, $\boldsymbol{f}_{\lfloor |\cup_{\ell=1}^{N}\Omega_\ell|/2 \rceil+1}, \ldots, \boldsymbol{f}_{|\cup_{\ell=1}^{N}\Omega_\ell|}$, thus the set of true change-points is divided into

two disjoint sets and each set of change-points occurs in disjoint sets of coordinates. In simulations, we compare the results of those three cases under the same level of cross-sectional sparsity, which allows us to see how the form of overlap between coordinates affects the results when the total number of coordinates experiencing changes at some points is fixed.

5.4.2 Simulation setting

Signals in scenario (S1)

We simulate data from the model (5.1) with the signal (5.3). As shown in Figure 5.4, we use 6 signals, (M1) bump, (M2) little.bump, (M3) three, (M4) teeth, (M5) extreme.teeth, (M6) blocks, that are specified below.

(M1) bump: T = 100, N = 2 change-points at t = 33, 66 with values between change-points 2, -2.

(M2) little.bump: T = 100, N = 2 change-points at t = 33, 66 with values between change-points 4/3, -4/3.

(M3) three: T = 200, N = 3 change-points at t = 50, 100, 150 with values between change-points 1, -1.5, 2.

(M4) teeth: T = 300, N = 9 change-points at t = 30, 60, 90, 120, 150, 180, 210, 240, 270 with values between change-points -2, 2, -2, 2, -2, 2, -2, 2, -2.

(M5) extreme.teeth: T = 500, N = 39 change-points at t = 13, 25, 38, 50, 63, 75, 88, 100, $\cdots$, 463, 475, 488 with values between change-points -3, 3, -3, 3, $\cdots$, 3, -3.

(M6) blocks: T = 1000, N = 11 change-points at t = 103, 134, 154, 236, 256, 410, 451, 666, 779, 799, 830 with values between change-points 1.464, -1.930, 1.298, -1.564, 1.830, -1.637, 1.168, 1.274, -1.535, 1.569, -1.937.

Signals in scenario (S2)

We simulate data from model (5.1) with the signal (5.4). Figure 5.5 shows 6 signals, (M1) one, (M2) wave, (M3) mix1, (M4) mix2, (M5) extreme.wave, (M6) lin.sgmts, that are specified below.

(M1) one: T = 100, N = 1 change-point at t = 50, with the corresponding jump size 0 and change in the slope -1/8 starting value for the intercept -1 and slope 1/16.

(M2) wave: T = 200, N = 9 change-points at t = 20, 40, 60, 80, 100, 120, 140, 160, 180, with the corresponding changes in the slope -0.5, 0.5, -0.5, 0.5, -0.5, 0.5, -0.5, 0.5, -0.5, starting value for the intercept -2 and slope 0.25.

(M3) mix1: T = 320, N = 7 change-points at t = 40, 80, 120, 160, 200, 240, 280, with the corresponding sizes of jump 1, -1, 0, -1, 1.5, -1, 0 and changes in the slope 0.2, -0.2, -0.2, 0.2, 0.2, -0.4, 0.4, starting value for the intercept -4 and slope 0.

(M4) mix2: T = 400, N = 9 change-points at t = 50, 100, 105, 150, 200, 250, 255, 300, 350, with the corresponding sizes of jump -4, 7, -5, -1, 1, -6.5, 5.5, 2.5, 0 and changes in the slope 0, 1/6, -1/4, 1/6, -1/6, 1/12, 0, 1/12, -1/6, starting value for the intercept 4 and slope 0.

(M5) extreme.wave: T = 500, N = 24 change-points at t = 20, 40, 60, 80, $\cdots$, 420, 440, 460, 480, with the corresponding changes in the slope -2/3, 2/3, -2/3, 2/3, $\cdots$, 2/3, -2/3, starting value for the intercept -4 and slope 1/3.

(M6) lin.sgmts: T = 500, N = 8 change-points at t = 100, 105, 200, 205, 300, 305, 400, 405, with the corresponding sizes of jump 6.5, -6.625, 6.5, -6.625, 6.5, -6.625, 6.5, -6.625 and changes in the slope 1/32, -1/32, 1/32, -1/32, 1/32, -1/32, 1/32, -1/32, starting value for the intercept -2 and slope 0.

5.4.3 Competing methods

We perform the HiTS procedure based on the parameter choice in Section 5.4.1 and compare the performance with that of the following competitors: Sparsified Binary Segmentation (**SBS**, Cho and Fryzlewicz (2015)) and Double Cusum (**DC**, Cho (2016)) implemented in the R package `hdbinseg` and Informative Sparse projection (**IS**, Wang and Samworth (2018)) available in the R package `InspectChangepoint`. The HiTS methodology is implemented in our GitHub repository (Maeng, 2019c). Regarding the tuning parameters for the competing methods, we follow the recommendation of each paper or the corresponding R package.

5.4.4 Simulation results

We run 100 simulations and the summary of the results can be found in Tables 5.2 - 5.13. We report Monte-Carlo estimates of the Mean Squared Error of the estimated signal defined as MSE=$\mathbb{E}\Big\{(1/T)\sum_{i=1}^{n}\sum_{t=1}^{T}(f_{i,t}-\hat{f}_{i,t})^2\Big\}$ and also give estimates of the scaled Hausdorff distance defined in (4.39). The small size of the Hausdorff distance indicates the better estimation of the change-point locations. We also report the empirical distribution of $\hat{N}-N$ where $\hat{N}$ is the estimated number of change-points and N is the true one. The average computation time in seconds is shown for each method. We note that R code for all simulations can be downloaded from our GitHub repository (Maeng, 2019c).

Result of scenario (S1)

The simulation results for all models and methods in scenario (S1) are summarised in Tables 5.2 - 5.10. The HiTS procedure has a particular advantage over other methods in terms of the estimation of the number and the locations of change-points when the level of sparsity is extreme (only one observation $\boldsymbol{X}_i$ includes true change-points i.e. n=100

and sparsity=0.01) in which other competing methods significantly underperform. The HiTS algorithm also outperforms in the case of "complete-overlap", while it is slightly less attractive when the level of sparsity is either "half-overlap" or "no-overlap" in which DC performs well and HiTS shows comparable results.

In all cases considered, all four methods including HiTS show better performances in (M2) with sparsity=0.1 compared to those in (M1) with sparsity=0.01. The only difference between models (M1) and (M2) is the jump size where the jump sizes of (M1) is 1.5 times larger than those of (M2). This implies that when a small number of change-points exist (which is 2 in models (M1) and (M2)), it is easier for all four methods to detect the weaker but denser signal than stronger but sparse signal. In model (M3) which includes three change-points with varying jump sizes, HiTS, SBS and IS give comparable performance to DC when sparsity level is 0.1 or 0.7, while DC exhibits better performance than others when sparsity is 0.01.

We see that HiTS is particularly attractive when relatively many (≥ 3) true change-points exist ((M4) and (M6)) or in the case of extremely frequent change-points in (M5). HiTS shows its robustness in estimating the number and the location of change-points in all sparsity levels and all sizes of n considered, except when sparsity is 0.01 in (M5) and (M6) in which no methods perform well. IS gives comparable performance to HiTS only in the case of "complete-overlap" and "half-overlap" with the sparsity level 0.7 while SBS and DC always significantly underestimate.

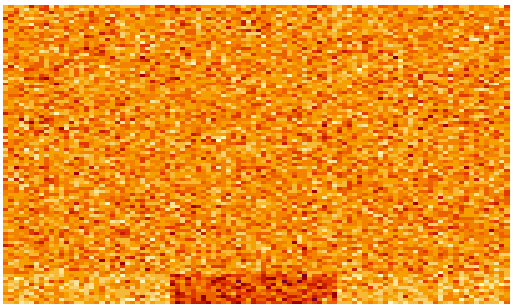
In all models (M1)-(M6), all four methods tend to show better performances in terms of the estimation of the number and the locations of change-points as the dimension n increases under any fixed level of overlap or as the level of overlap increases in order of "no-overlap", "half-overlap" and "complete-overlap" under any fixed n. With respect to computation time, HiTS is very fast (less than 1.5 seconds) in all cases and the computation time does not increase proportional to the dimension, while SBS, DC and

IS are much slower than HiTS especially when either the dimension n is larger than equal to 300 or the length of data sequences T is larger than equal to 500.

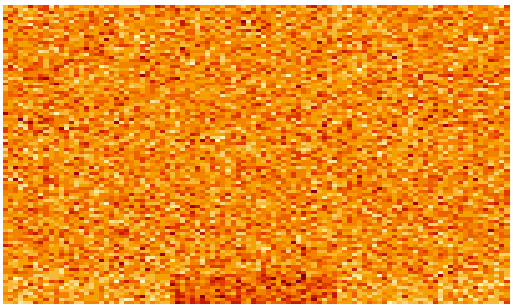
Result of scenario (S2)

The summary of the simulation results for all models and methods in scenario (S2) can be found in Tables 5.11 - 5.13. As is in scenario (S1), HiTS performs well not only in single change-point but also in multiple and/or frequent change-points. In general, the HiTS procedure shows better performance as the level of overlap increases in order of "no-overlap", "half-overlap" and "complete-overlap" under any fixed n and a fixed sparsity level.

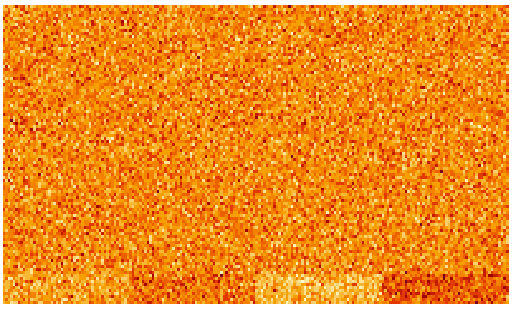
When the length of data sequences T is relatively larger than the dimension of the data n, in any model, the estimation of the number of change-points of HiTS tends to be improved when the sparsity level increases in the cases of "complete-overlap" and "half-overlap", but the tendency is not clear in "no-overlap" case. When the level of sparsity is extreme (i.e. n=100 and sparsity=0.01), HiTS relatively underperforms in (M2)-(M4) in all cases of overlap.



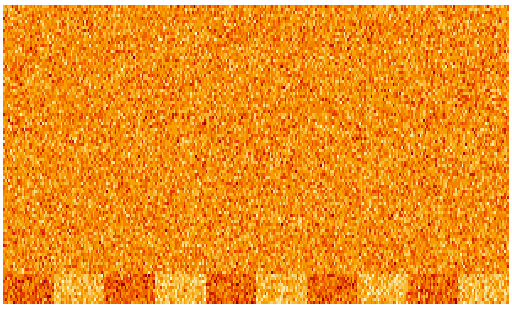
(a) (M1) bump



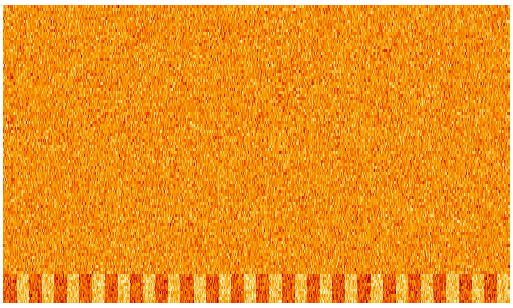
(b) (M2) little.bump



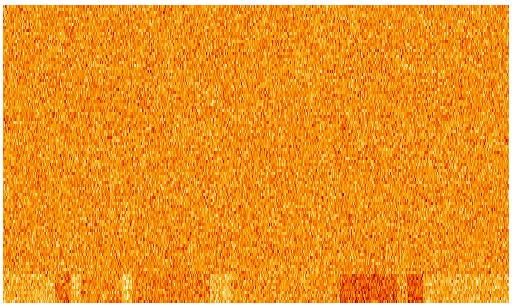
(c) (M3) three



(d) (M4) teeth



(e) (M5) extreme.teeth



(f) (M6) blocks

Fig. 5.4 Examples of data with its underlying signal studied in Section 5.4.2 in scenario (S1). (a)-(f) visualisation of the data matrix $\boldsymbol{X}$ when n=100, sparsity=0.1 and "complete-overlap" case.

Table 5.2 Distribution of $\hat{N} - N$ for models (M1)-(M6) and all methods over 100 simulation runs in the "complete-overlap" case with $n = 100$ in scenario (S1). Also the average MSE (Mean Squared Error) of the estimated signal $\hat{\boldsymbol{f}}$, the average Hausdorff distance d_H and the average computational time in seconds using an Intel Core i9 3.6 GHz CPU with 8 GB of RAM, all over 100 simulations. Bold: methods within 10% of the highest empirical frequency of $\hat{N} - N = 0$ or within 10% of the lowest empirical average $d_H(\times 10^2)$.

			$\hat{N} - N$												$\hat{N} - N$									
Model	sparsity	Method	≤-3	-2	-1	0	1	2	≥3	MSE	$d_H(\times 10^2)$	time	Model	sparsity	Method	≤-3	-2	-1	0	1	2	≥3	MSE	$d_H(\times 10^2)$	time
(M1)	0.01	HiTS	0	16	2	**77**	2	2	1	3.10	**10.29**	0.16	(M4)	0.01	HiTS	56	27	0	**17**	0	0	0	2.53	**22.79**	0.31
		SBS	0	100	0	0	0	0	0	1.88	34.00	1.17			SBS	100	0	0	0	0	0	0	1.34	50.00	1.55
		DC	0	65	0	34	1	0	0	2.29	22.66	2.84			DC	100	0	0	0	0	0	0	1.47	45.47	3.89
		IS	0	89	3	5	3	0	0	2.06	31.69	0.36			IS	100	0	0	0	0	0	0	1.38	48.52	0.73
	0.1	HiTS	0	0	0	**92**	3	4	1	3.50	**1.57**	0.19		0.1	HiTS	0	0	0	**98**	1	1	0	3.89	**0.68**	0.31
		SBS	0	67	0	33	0	0	0	7.68	23.05	1.18			SBS	100	0	0	0	0	0	0	10.34	50.00	1.56
		DC	0	11	0	83	6	0	0	3.97	4.63	2.87			DC	100	0	0	0	0	0	0	10.34	50.00	5.20
		IS	0	0	0	**88**	10	1	1	3.21	**1.40**	0.36			IS	0	0	0	79	15	5	1	3.56	**0.87**	0.78
	0.7	HiTS	0	0	0	95	2	3	0	3.15	0.56	0.20		0.7	HiTS	0	0	0	**99**	0	1	0	3.43	**0.07**	0.31
		SBS	0	0	0	**100**	0	0	0	3.01	**0.01**	1.18			SBS	100	0	0	0	0	0	0	70.34	50.00	1.57
		DC	0	0	0	98	2	0	0	3.01	0.25	2.84			DC	100	0	0	0	0	0	0	70.34	50.00	5.15
		IS	0	0	0	94	5	1	0	3.08	0.68	0.37			IS	0	0	0	**90**	7	3	0	3.42	**0.31**	0.78
(M2)	0.01	HiTS	0	62	5	**30**	3	0	0	2.07	**26.23**	0.16	(M5)	0.01	HiTS	100	0	0	0	0	0	0	4.69	**20.11**	0.37
		SBS	0	100	0	0	0	0	0	1.39	34.00	1.18			SBS	100	0	0	0	0	0	0	2.45	49.88	1.92
		DC	0	92	0	8	0	0	0	1.52	31.40	2.78			DC	100	0	0	0	0	0	0	2.45	49.75	5.12
		IS	0	96	4	0	0	0	0	1.44	33.94	0.36			IS	100	0	0	0	0	0	0	2.45	49.78	1.10
	0.1	HiTS	0	1	1	**90**	4	3	1	3.84	**5.06**	0.15		0.1	HiTS	0	0	0	**99**	0	1	0	8.55	**0.28**	0.39
		SBS	0	93	1	6	0	0	0	4.86	32.09	1.18			SBS	100	0	0	0	0	0	0	22.66	50.00	1.95
		DC	0	73	0	22	5	0	0	4.59	26.07	2.86			DC	100	0	0	0	0	0	0	22.66	50.00	7.14
		IS	0	0	0	**84**	14	1	1	3.34	**2.07**	0.36			IS	0	0	0	61	32	7	0	8.19	**0.34**	1.34
	0.7	HiTS	0	0	0	**91**	6	3	0	3.73	**1.21**	0.17		0.7	HiTS	0	0	0	**100**	0	0	0	8.00	**0.00**	0.39
		SBS	0	35	0	65	0	0	0	12.25	12.17	1.19			SBS	100	0	0	0	0	0	0	157.45	50.00	1.99
		DC	0	13	0	85	2	0	0	6.37	4.71	2.84			DC	100	0	0	0	0	0	0	157.45	50.00	7.14
		IS	0	0	0	**90**	9	1	0	3.14	**1.07**	0.36			IS	0	0	0	**91**	9	0	0	8.02	**0.07**	1.35
(M3)	0.01	HiTS	4	52	29	15	0	0	0	1.58	23.57	0.27	(M6)	0.01	HiTS	95	2	1	**2**	0	0	0	0.81	**14.75**	0.49
		SBS	69	27	4	0	0	0	0	1.21	42.35	1.36			SBS	100	0	0	0	0	0	0	0.62	44.69	2.79
		DC	0	39	32	**29**	0	0	0	1.65	**18.77**	3.27			DC	100	0	0	0	0	0	0	0.66	22.50	8.61
		IS	31	54	13	1	0	0	1	1.37	32.56	0.55			IS	100	0	0	0	0	0	0	0.60	30.60	2.00
	0.1	HiTS	0	0	13	87	0	0	0	2.49	5.53	0.26		0.1	HiTS	1	15	10	**74**	0	0	0	1.51	**1.22**	0.50
		SBS	0	5	17	78	0	0	0	2.31	5.92	1.36			SBS	100	0	0	0	0	0	0	3.69	32.23	2.83
		DC	0	0	1	**98**	1	0	0	2.14	**0.92**	3.26			DC	100	0	0	0	0	0	0	2.88	25.80	12.21
		IS	0	0	0	96	4	0	0	2.08	**0.44**	0.56			IS	0	1	0	64	20	11	4	1.33	**0.78**	2.06
	0.7	HiTS	0	0	0	98	1	1	0	2.37	0.47	0.26		0.7	HiTS	0	0	0	**100**	0	0	0	1.33	**0.08**	0.50
		SBS	0	0	0	**100**	0	0	0	2.08	**0.06**	1.35			SBS	100	0	0	0	0	0	0	18.05	25.60	2.95
		DC	0	0	0	99	1	0	0	2.02	**0.08**	3.26			DC	100	0	0	0	0	0	0	19.26	26.77	12.34
		IS	0	0	0	96	4	0	0	2.04	0.26	0.56			IS	0	0	0	80	14	6	0	1.23	**0.55**	2.05

Table 5.3 Distribution of $\hat{N} - N$ for models (M1)-(M6) and all methods over 100 simulation runs in the "complete-overlap" case with $n = 300$ in scenario (S1). Also the average MSE (Mean Squared Error) of the estimated signal $\hat{\boldsymbol{f}}$, the average Hausdorff distance d_H and the average computational time in seconds using an Intel Core i9 3.6 GHz CPU with 8 GB of RAM, all over 100 simulations. Bold: methods within 10% of the highest empirical frequency of $\hat{N} - N = 0$ or within 10% of the lowest empirical average $d_H(\times 10^2)$.

			$\hat{N} - N$												$\hat{N} - N$									
Model	sparsity	Method	≤-3	-2	-1	0	1	2	≥3	MSE	$d_H(\times 10^2)$	time	Model	sparsity	Method	≤-3	-2	-1	0	1	2	≥3	MSE	$d_H(\times 10^2)$	time
(M1)	0.01	HiTS	0	4	2	**89**	3	2	0	9.61	**6.50**	0.37	(M4)	0.01	HiTS	13	19	0	**68**	0	0	0	9.89	**6.99**	0.52
		SBS	0	98	1	1	0	0	0	5.70	33.69	1.77			SBS	100	0	0	0	0	0	0	4.00	49.74	2.89
		DC	0	53	0	47	0	0	0	7.33	18.77	6.11			DC	100	0	0	0	0	0	0	4.55	43.55	8.60
		IS	0	78	1	17	2	2	0	6.64	27.54	3.45			IS	100	0	0	0	0	0	0	4.38	45.42	5.14
	0.1	HiTS	0	0	0	**94**	1	5	0	10.00	**1.17**	0.36		0.1	HiTS	0	0	0	**100**	0	0	0	10.86	**0.43**	0.53
		SBS	0	22	0	78	0	0	0	13.72	7.61	1.78			SBS	100	0	0	0	0	0	0	30.99	50.00	2.90
		DC	0	2	0	**96**	2	0	0	9.81	**1.14**	6.21			DC	100	0	0	0	0	0	0	30.99	50.00	9.92
		IS	0	0	0	75	21	4	0	10.04	3.15	3.55			IS	0	0	0	76	13	8	3	10.52	**0.89**	5.68
	0.7	HiTS	0	0	0	96	0	4	0	9.59	0.47	0.35		0.7	HiTS	0	0	0	**100**	0	0	0	10.14	**0.01**	0.53
		SBS	0	0	0	**100**	0	0	0	9.01	**0.00**	1.78			SBS	100	0	0	0	0	0	0	210.99	50.00	2.95
		DC	0	0	0	**99**	1	0	0	9.04	**0.11**	6.22			DC	100	0	0	0	0	0	0	210.99	50.00	9.90
		IS	0	0	0	83	16	1	0	9.64	2.04	3.55			IS	0	0	0	84	10	5	1	10.31	**0.58**	5.69
(M2)	0.01	HiTS	0	64	4	**32**	0	0	0	5.96	**25.65**	0.38	(M5)	0.01	HiTS	85	7	0	**8**	0	0	0	19.72	**9.53**	0.66
		SBS	0	99	1	0	0	0	0	4.19	34.00	1.77			SBS	100	0	0	0	0	0	0	7.37	49.02	3.91
		DC	0	91	2	7	0	0	0	4.57	31.74	6.05			DC	100	0	0	0	0	0	0	7.34	49.75	11.53
		IS	0	87	3	5	4	1	0	5.01	31.42	3.44			IS	100	0	0	0	0	0	0	7.42	49.35	6.70
	0.1	HiTS	0	0	0	**94**	1	5	0	10.63	**2.82**	0.37		0.1	HiTS	0	0	0	**100**	0	0	0	24.58	**0.13**	0.67
		SBS	0	63	0	37	0	0	0	13.05	22.09	1.78			SBS	100	0	0	0	0	0	0	68.02	48.84	3.98
		DC	0	55	0	44	1	0	0	12.46	19.27	6.13			DC	100	0	0	0	0	0	0	67.99	50.00	13.65
		IS	0	0	0	75	21	4	0	10.14	**3.24**	3.55			IS	0	0	0	58	17	14	11	24.72	**0.42**	9.36
	0.7	HiTS	0	0	0	95	0	5	0	9.97	0.67	0.36		0.7	HiTS	0	0	0	**100**	0	0	0	24.19	**0.01**	0.68
		SBS	0	6	0	94	0	0	0	14.06	2.16	1.78			SBS	100	0	0	0	0	0	0	472.34	50.00	4.05
		DC	0	0	0	**99**	1	0	0	9.04	**0.11**	6.22			DC	100	0	0	0	0	0	0	472.34	50.00	13.64
		IS	0	0	0	81	17	2	0	9.74	2.32	3.55			IS	0	0	0	**97**	2	1	0	24.11	**0.03**	9.38
(M3)	0.01	HiTS	0	26	49	25	0	0	0	5.10	20.16	0.44	(M6)	0.01	HiTS	80	19	0	**1**	0	0	0	2.70	**9.89**	0.93
		SBS	56	32	10	2	0	0	0	3.76	38.68	2.34			SBS	100	0	0	0	0	0	0	1.87	43.57	6.31
		DC	0	6	36	**58**	0	0	0	5.55	**11.32**	7.19			DC	100	0	0	0	0	0	0	2.12	20.95	19.50
		IS	0	35	37	20	5	3	0	5.12	19.21	4.41			IS	99	1	0	0	0	0	0	2.00	22.46	10.78
	0.1	HiTS	0	0	3	95	1	1	0	6.82	2.28	0.44		0.1	HiTS	0	6	0	**94**	0	0	0	4.14	**0.66**	0.94
		SBS	0	0	2	98	0	0	0	6.16	0.70	2.34			SBS	100	0	0	0	0	0	0	8.70	25.87	6.41
		DC	0	0	0	**100**	0	0	0	6.07	**0.13**	7.19			DC	100	0	0	0	0	0	0	8.63	25.80	23.75
		IS	0	0	0	86	9	5	0	6.35	1.32	4.47			IS	0	0	0	80	13	6	1	3.74	**0.50**	11.29
	0.7	HiTS	0	0	0	98	1	1	0	6.79	0.42	0.44		0.7	HiTS	0	0	0	**100**	0	0	0	3.80	**0.05**	0.93
		SBS	0	0	0	**100**	0	0	0	6.02	**0.00**	2.35			SBS	100	0	0	0	0	0	0	53.29	25.50	6.77
		DC	0	0	0	**100**	0	0	0	6.01	**0.00**	7.19			DC	100	0	0	0	0	0	0	55.02	25.80	23.94
		IS	0	0	0	88	8	4	0	6.29	1.08	4.48			IS	0	0	0	85	12	2	1	3.68	**0.36**	11.29

Table 5.4 Distribution of $\hat{N} - N$ for models (M1)-(M6) and all methods over 100 simulation runs in the "complete-overlap" case with $n = 500$ in scenario (S1). Also the average MSE (Mean Squared Error) of the estimated signal $\hat{\boldsymbol{f}}$, the average Hausdorff distance d_H and the average computational time in seconds using an Intel Core i9 3.6 GHz CPU with 8 GB of RAM, all over 100 simulations. Bold: methods within 10% of the highest empirical frequency of $\hat{N} - N = 0$ or within 10% of the lowest empirical average $d_H(\times 10^2)$.

			$\hat{N}-N$												$\hat{N}-N$									
Model	sparsity	Method	≤-3	-2	-1	0	1	2	≥3	MSE	$d_H(\times 10^2)$	time	Model	sparsity	Method	≤-3	-2	-1	0	1	2	≥3	MSE	$d_H(\times 10^2)$	time
(M1)	0.01	HiTS	0	4	0	**86**	6	4	0	16.25	**5.88**	0.48	(M4)	0.01	HiTS	6	12	0	**82**	0	0	0	16.97	**4.03**	0.73
		SBS	0	95	1	4	0	0	0	9.68	32.68	2.34			SBS	100	0	0	0	0	0	0	6.74	49.06	4.07
		DC	0	40	0	59	1	0	0	12.98	14.61	9.51			DC	100	0	0	0	0	0	0	8.05	40.82	13.53
		IS	0	76	0	17	4	3	0	11.33	26.83	13.08			IS	96	2	1	1	0	0	0	8.82	37.44	17.12
	0.1	HiTS	0	0	0	89	7	4	0	16.64	1.35	0.48		0.1	HiTS	0	0	0	**100**	0	0	0	17.90	**0.43**	0.73
		SBS	0	9	0	91	0	0	0	18.25	3.16	2.35			SBS	100	0	0	0	0	0	0	51.68	49.74	4.12
		DC	0	2	0	**97**	1	0	0	15.81	**0.79**	9.49			DC	100	0	0	0	0	0	0	51.67	50.00	14.99
		IS	0	0	0	82	14	3	1	16.23	2.27	13.47			IS	0	0	0	66	19	12	3	17.62	**1.28**	19.32
	0.7	HiTS	0	0	0	91	5	4	0	16.04	0.95	0.47		0.7	HiTS	0	0	0	**100**	0	0	0	16.81	**0.01**	0.73
		SBS	0	0	0	**100**	0	0	0	14.97	**0.00**	2.35			SBS	100	0	0	0	0	0	0	351.67	50.00	4.18
		DC	0	0	0	**100**	0	0	0	14.97	**0.00**	9.56			DC	100	0	0	0	0	0	0	351.67	50.00	14.97
		IS	0	0	0	91	8	1	0	15.53	1.21	13.45			IS	0	0	0	87	12	1	0	16.90	**0.48**	19.36
(M2)	0.01	HiTS	0	52	4	**41**	2	1	0	11.11	**21.81**	0.49	(M5)	0.01	HiTS	57	24	1	**18**	0	0	0	37.32	**5.26**	0.94
		SBS	0	97	0	3	0	0	0	7.20	33.07	2.35			SBS	100	0	0	0	0	0	0	12.27	49.01	5.75
		DC	0	92	1	6	1	0	0	7.61	31.87	9.52			DC	100	0	0	0	0	0	0	12.22	50.00	18.23
		IS	0	77	4	13	5	1	0	9.16	28.86	13.07			IS	100	0	0	0	0	0	0	12.39	49.18	20.52
	0.1	HiTS	0	0	0	**90**	6	4	0	17.64	**2.81**	0.49		0.1	HiTS	0	0	0	**100**	0	0	0	40.41	**0.08**	0.97
		SBS	0	51	0	49	0	0	0	20.22	17.71	2.35			SBS	100	0	0	0	0	0	0	113.32	49.75	5.87
		DC	0	56	0	42	2	0	0	20.85	19.71	9.47			DC	100	0	0	0	0	0	0	113.31	50.00	20.63
		IS	0	0	0	83	12	3	2	16.33	**2.13**	13.45			IS	0	0	0	46	21	15	18	41.40	**0.55**	30.63
	0.7	HiTS	0	0	0	91	4	5	0	16.81	1.10	0.47		0.7	HiTS	0	0	0	**100**	0	0	0	39.97	**0.00**	0.98
		SBS	0	2	0	**98**	0	0	0	17.65	0.70	2.35			SBS	100	0	0	0	0	0	0	787.23	50.00	5.95
		DC	0	0	0	**99**	1	0	0	15.02	**0.16**	9.57			DC	100	0	0	0	0	0	0	787.23	50.00	20.57
		IS	0	0	0	86	11	3	0	15.90	1.75	13.46			IS	0	0	0	**95**	5	0	0	40.03	**0.04**	30.47
(M3)	0.01	HiTS	0	11	50	38	1	0	0	9.08	16.69	0.61	(M6)	0.01	HiTS	68	24	2	**6**	0	0	0	4.71	**8.84**	1.37
		SBS	27	53	14	6	0	0	0	6.75	30.48	3.26			SBS	100	0	0	0	0	0	0	3.08	42.94	10.24
		DC	0	5	22	**72**	1	0	0	9.57	**7.81**	11.29			DC	100	0	0	0	0	0	0	3.55	20.10	31.15
		IS	0	10	31	48	8	3	0	9.55	11.79	15.46			IS	92	3	2	2	1	0	0	3.84	18.78	30.22
	0.1	HiTS	0	0	2	96	1	1	0	11.27	2.04	0.61		0.1	HiTS	0	0	1	**99**	0	0	0	6.62	**0.33**	1.38
		SBS	0	0	0	**100**	0	0	0	10.08	**0.12**	3.27			SBS	100	0	0	0	0	0	0	14.27	25.43	10.45
		DC	0	0	0	**100**	0	0	0	10.01	**0.01**	11.26			DC	100	0	0	0	0	0	0	14.49	25.99	35.42
		IS	0	0	0	83	12	4	1	10.64	1.56	15.60			IS	0	0	0	64	18	14	4	6.38	**1.03**	31.91
	0.7	HiTS	0	0	0	99	0	1	0	10.83	0.27	0.61		0.7	HiTS	0	0	0	**100**	0	0	0	6.22	**0.03**	1.35
		SBS	0	0	0	**100**	0	0	0	10.00	**0.00**	3.27			SBS	100	0	0	0	0	0	0	89.60	25.60	10.71
		DC	0	0	0	**100**	0	0	0	10.00	**0.00**	11.25			DC	100	0	0	0	0	0	0	90.77	25.60	34.53
		IS	0	0	0	92	6	1	1	10.31	0.74	15.59			IS	0	0	0	70	20	6	4	6.26	**0.82**	31.59

Table 5.5 Distribution of $\hat{N} - N$ for models (M1)-(M6) and all methods over 100 simulation runs in the "half-overlap" case with $n = 100$ in scenario (S1). Also the average MSE (Mean Squared Error) of the estimated signal $\hat{\boldsymbol{f}}$, the average Hausdorff distance d_H and the average computational time in seconds using an Intel Core i9 3.6 GHz CPU with 8 GB of RAM, all over 100 simulations. Bold: methods within 10% of the highest empirical frequency of $\hat{N} - N = 0$ or within 10% of the lowest empirical average $d_H(\times 10^2)$.

			$\hat{N} - N$												$\hat{N} - N$									
Model	sparsity	Method	≤-3	-2	-1	0	1	2	≥3	MSE	$d_H(\times 10^2)$	time	Model	sparsity	Method	≤-3	-2	-1	0	1	2	≥3	MSE	$d_H(\times 10^2)$	time
(M1)	0.01	HiTS	0	16	2	**77**	2	2	1	3.10	**10.29**	0.16	(M4)	0.01	HiTS	56	27	0	**17**	0	0	0	2.53	**22.79**	0.30
		SBS	0	100	0	0	0	0	0	1.88	34.00	1.19			SBS	100	0	0	0	0	0	0	1.34	50.00	1.53
		DC	0	63	0	36	1	0	0	2.32	22.11	2.88			DC	100	0	0	0	0	0	0	1.47	45.11	3.86
		IS	0	89	3	5	3	0	0	2.06	31.69	0.36			IS	100	0	0	0	0	0	0	1.38	48.52	0.74
	0.1	HiTS	0	0	0	92	3	4	1	3.52	2.04	0.15		0.1	HiTS	0	0	0	**99**	0	1	0	3.92	**1.08**	0.30
		SBS	0	0	4	96	0	0	0	3.16	1.58	1.22			SBS	100	0	0	0	0	0	0	8.12	40.33	1.54
		DC	0	0	0	**98**	2	0	0	3.15	**0.67**	2.89			DC	16	84	0	0	0	0	0	4.47	12.27	3.94
		IS	0	0	0	91	8	0	1	3.14	1.14	0.37			IS	0	0	0	65	25	9	1	3.67	**1.17**	0.78
	0.7	HiTS	0	0	0	94	3	3	0	3.25	0.67	0.13		0.7	HiTS	0	0	0	**99**	0	1	0	3.51	**0.13**	0.30
		SBS	0	0	0	**100**	0	0	0	3.04	**0.02**	1.24			SBS	100	0	0	0	0	0	0	49.44	29.92	1.56
		DC	0	0	0	99	1	0	0	2.99	0.11	2.89			DC	100	0	0	0	0	0	0	51.76	35.40	5.11
		IS	0	0	0	94	5	0	1	3.09	0.67	0.37			IS	0	0	0	89	8	3	0	3.43	**0.34**	0.78
(M2)	0.01	HiTS	0	62	5	**30**	3	0	0	2.07	**26.23**	0.15	(M5)	0.01	HiTS	100	0	0	0	0	0	0	4.69	**20.11**	0.37
		SBS	0	100	0	0	0	0	0	1.39	34.00	1.19			SBS	100	0	0	0	0	0	0	2.45	49.73	1.91
		DC	0	94	0	5	1	0	0	1.50	32.16	2.78			DC	100	0	0	0	0	0	0	2.45	49.75	5.11
		IS	0	96	4	0	0	0	0	1.44	33.94	0.36			IS	100	0	0	0	0	0	0	2.45	49.78	1.11
	0.1	HiTS	0	1	11	78	7	2	1	3.76	8.41	0.12		0.1	HiTS	0	7	0	**92**	0	1	0	8.64	**0.54**	0.38
		SBS	0	0	29	71	0	0	0	3.25	10.26	1.22			SBS	100	0	0	0	0	0	0	17.25	25.00	1.95
		DC	0	5	10	84	1	0	0	3.33	6.38	2.88			DC	100	0	0	0	0	0	0	17.25	25.00	6.70
		IS	0	0	0	**90**	9	0	1	3.21	**1.61**	0.37			IS	11	5	0	49	27	7	1	8.36	**1.95**	1.29
	0.7	HiTS	0	0	0	92	5	3	0	3.83	1.52	0.13		0.7	HiTS	0	0	0	**100**	0	0	0	8.01	**0.00**	0.38
		SBS	0	0	0	**100**	0	0	0	3.13	**0.16**	1.23			SBS	100	0	0	0	0	0	0	118.33	25.00	1.99
		DC	0	0	0	99	1	0	0	3.01	**0.12**	2.89			DC	100	0	0	0	0	0	0	118.33	25.00	7.03
		IS	0	0	0	93	6	0	1	3.10	0.74	0.37			IS	0	0	0	87	13	0	0	8.03	**0.11**	1.29
(M3)	0.01	HiTS	4	52	29	15	0	0	0	1.58	23.57	0.26	(M6)	0.01	HiTS	95	2	1	**2**	0	0	0	0.81	**14.75**	0.48
		SBS	69	27	4	0	0	0	0	1.21	42.38	1.35			SBS	100	0	0	0	0	0	0	0.62	45.10	2.77
		DC	0	42	31	**27**	0	0	0	1.63	**19.20**	3.23			DC	100	0	0	0	0	0	0	0.66	22.40	8.57
		IS	31	54	13	1	0	0	1	1.37	32.56	0.55			IS	100	0	0	0	0	0	0	0.60	30.60	1.95
	0.1	HiTS	0	0	13	87	0	0	0	2.50	5.59	0.26		0.1	HiTS	14	24	6	**56**	0	0	0	1.48	**2.59**	0.48
		SBS	0	11	16	73	0	0	0	2.61	7.78	1.35			SBS	100	0	0	0	0	0	0	2.94	22.27	2.81
		DC	0	0	0	**99**	1	0	0	2.17	**0.85**	3.23			DC	100	0	0	0	0	0	0	1.55	5.06	8.31
		IS	0	0	0	95	5	0	0	2.09	**0.48**	0.56			IS	0	31	3	42	15	7	2	1.31	**2.48**	2.01
	0.7	HiTS	0	0	0	98	1	1	0	2.33	0.46	0.26		0.7	HiTS	0	0	0	**100**	0	0	0	1.36	**0.14**	0.48
		SBS	0	0	0	**100**	0	0	0	2.39	0.28	1.35			SBS	100	0	0	0	0	0	0	11.16	14.24	2.93
		DC	0	0	0	99	1	0	0	2.02	**0.08**	3.23			DC	100	0	0	0	0	0	0	5.42	5.10	11.91
		IS	0	0	0	96	4	0	0	2.04	0.26	0.56			IS	0	0	0	79	16	5	0	1.24	**0.56**	2.01

Table 5.6 Distribution of $\hat{N} - N$ for models (M1)-(M6) and all methods over 100 simulation runs in the "half-overlap" case with $n = 300$ in scenario (S1). Also the average MSE (Mean Squared Error) of the estimated signal $\hat{\boldsymbol{f}}$, the average Hausdorff distance d_H and the average computational time in seconds using an Intel Core i9 3.6 GHz CPU with 8 GB of RAM, all over 100 simulations. Bold: methods within 10% of the highest empirical frequency of $\hat{N} - N = 0$ or within 10% of the lowest empirical average $d_H(\times 10^2)$.

			$\hat{N} - N$												$\hat{N} - N$									
Model	sparsity	Method	≤-3	-2	-1	0	1	2	≥3	MSE	$d_H(\times 10^2)$	time	Model	sparsity	Method	≤-3	-2	-1	0	1	2	≥3	MSE	$d_H(\times 10^2)$	time
(M1)	0.01	HiTS	0	0	21	76	1	2	0	9.06	11.45	0.35	(M4)	0.01	HiTS	60	22	0	**18**	0	0	0	7.48	**18.25**	0.51
		SBS	0	19	63	18	0	0	0	6.97	27.26	1.76			SBS	100	0	0	0	0	0	0	3.35	47.67	2.88
		DC	0	1	13	**86**	0	0	0	8.75	**5.46**	6.17			DC	90	10	0	0	0	0	0	6.04	22.23	8.56
		IS	0	1	64	22	12	1	0	8.09	24.72	3.50			IS	100	0	0	0	0	0	0	4.40	30.23	5.17
	0.1	HiTS	0	0	0	94	1	5	0	10.06	1.52	0.34		0.1	HiTS	0	0	0	**100**	0	0	0	10.95	**0.67**	0.52
		SBS	0	0	0	**100**	0	0	0	9.13	**0.11**	1.77			SBS	100	0	0	0	0	0	0	23.72	35.68	2.89
		DC	0	0	0	**100**	0	0	0	9.03	**0.02**	6.16			DC	100	0	0	0	0	0	0	21.53	29.85	9.88
		IS	0	0	0	76	20	4	0	9.98	2.99	3.55			IS	0	0	0	73	15	9	3	10.60	**0.98**	5.61
	0.7	HiTS	0	0	0	96	0	3	1	9.70	0.49	0.34		0.7	HiTS	0	0	0	**100**	0	0	0	10.35	**0.05**	0.51
		SBS	0	0	0	**100**	0	0	0	9.01	**0.00**	1.77			SBS	100	0	0	0	0	0	0	143.42	29.97	2.93
		DC	0	0	0	**100**	0	0	0	9.01	**0.00**	6.16			DC	100	0	0	0	0	0	0	163.76	42.20	9.84
		IS	0	0	0	84	14	2	0	9.64	1.94	3.54			IS	0	0	0	83	8	7	2	10.37	**0.61**	5.61
(M2)	0.01	HiTS	0	33	52	15	0	0	0	6.16	29.78	0.35	(M5)	0.01	HiTS	98	0	0	**2**	0	0	0	14.97	**17.01**	0.65
		SBS	0	37	60	3	0	0	0	5.65	32.08	1.76			SBS	100	0	0	0	0	0	0	5.71	25.11	3.92
		DC	0	22	51	**27**	0	0	0	6.59	**24.69**	6.15			DC	100	0	0	0	0	0	0	5.71	25.00	11.41
		IS	0	42	48	9	1	0	0	5.83	32.14	3.46			IS	100	0	0	0	0	0	0	5.80	24.99	6.66
	0.1	HiTS	0	0	6	89	1	4	0	10.53	4.97	0.35		0.1	HiTS	0	0	0	**100**	0	0	0	24.73	**0.21**	0.67
		SBS	0	0	0	**100**	0	0	0	9.15	**0.33**	1.77			SBS	100	0	0	0	0	0	0	51.75	25.00	3.97
		DC	0	0	2	**98**	0	0	0	9.29	1.41	6.17			DC	100	0	0	0	0	0	0	51.74	25.00	13.53
		IS	0	0	0	76	20	4	0	9.99	2.99	3.55			IS	0	0	0	57	18	12	13	24.77	**0.44**	8.94
	0.7	HiTS	0	0	0	94	1	5	0	10.43	1.03	0.35		0.7	HiTS	0	0	0	**100**	0	0	0	24.15	**0.01**	0.67
		SBS	0	0	0	**100**	0	0	0	9.01	**0.00**	1.77			SBS	100	0	0	0	0	0	0	355.01	25.00	4.05
		DC	0	0	0	**100**	0	0	0	9.01	**0.00**	6.16			DC	100	0	0	0	0	0	0	355.01	25.00	13.51
		IS	0	0	0	81	18	1	0	9.71	2.33	3.54			IS	0	0	0	**93**	6	0	1	24.15	**0.07**	8.92
(M3)	0.01	HiTS	6	17	49	28	0	0	0	5.07	20.89	0.43	(M6)	0.01	HiTS	93	7	0	0	0	0	0	2.43	**8.94**	0.91
		SBS	27	40	27	6	0	0	0	4.07	30.88	2.32			SBS	100	0	0	0	0	0	0	1.46	25.53	6.27
		DC	1	6	43	**49**	1	0	0	5.43	**13.56**	7.21			DC	100	0	0	0	0	0	0	2.49	**8.78**	19.11
		IS	43	21	23	10	2	1	0	4.12	33.16	4.34			IS	100	0	0	0	0	0	0	1.71	22.35	10.39
	0.1	HiTS	0	0	3	95	1	1	0	6.89	2.35	0.43		0.1	HiTS	1	15	2	**82**	0	0	0	4.11	**1.29**	0.91
		SBS	0	0	0	**100**	0	0	0	6.48	0.73	2.33			SBS	100	0	0	0	0	0	0	7.46	18.47	6.40
		DC	0	0	0	**99**	1	0	0	6.12	**0.22**	7.16			DC	100	0	0	0	0	0	0	4.51	5.14	19.11
		IS	0	0	0	86	9	5	0	6.36	1.34	4.47			IS	0	2	1	73	19	4	1	3.76	**0.66**	10.95
	0.7	HiTS	0	0	0	97	2	1	0	6.75	0.44	0.43		0.7	HiTS	0	0	0	**100**	0	0	0	3.87	**0.10**	0.91
		SBS	0	0	0	**100**	0	0	0	6.48	**0.12**	2.33			SBS	100	0	0	0	0	0	0	33.15	14.43	6.74
		DC	0	0	0	**100**	0	0	0	6.01	**0.00**	7.15			DC	100	0	0	0	0	0	0	16.35	5.20	23.30
		IS	0	0	0	86	10	4	0	6.32	1.32	4.47			IS	0	0	0	83	13	3	1	3.69	**0.42**	10.99

Table 5.7 Distribution of $\hat{N} - N$ for models (M1)-(M6) and all methods over 100 simulation runs in the "half-overlap" case with $n = 500$ in scenario (S1). Also the average MSE (Mean Squared Error) of the estimated signal $\hat{\boldsymbol{f}}$, the average Hausdorff distance d_H and the average computational time in seconds using an Intel Core i9 3.6 GHz CPU with 8 GB of RAM, all over 100 simulations. Bold: methods within 10% of the highest empirical frequency of $\hat{N} - N = 0$ or within 10% of the lowest empirical average $d_H(\times 10^2)$.

			$\hat{N}-N$												$\hat{N}-N$									
Model	sparsity	Method	≤-3	-2	-1	0	1	2	≥3	MSE	$d_H(\times 10^2)$	time	Model	sparsity	Method	≤-3	-2	-1	0	1	2	≥3	MSE	$d_H(\times 10^2)$	time
(M1)	0.01	HiTS	0	0	8	83	5	4	0	16.10	7.58	0.46	(M4)	0.01	HiTS	17	18	2	**63**	0	0	0	15.92	**7.18**	0.70
		SBS	0	12	36	52	0	0	0	13.45	16.24	2.33			SBS	100	0	0	0	0	0	0	6.09	47.69	4.06
		DC	0	0	2	**98**	0	0	0	15.02	**1.47**	9.47			DC	74	26	0	0	0	0	0	11.68	17.36	13.53
		IS	0	0	2	77	17	3	1	16.38	3.44	13.43			IS	99	1	0	0	0	0	0	8.47	33.05	17.17
	0.1	HiTS	0	0	0	89	7	3	1	16.88	1.70	0.46		0.1	HiTS	0	0	0	**100**	0	0	0	17.90	**0.66**	0.72
		SBS	0	0	0	**100**	0	0	0	14.97	**0.00**	2.34			SBS	100	0	0	0	0	0	0	39.12	33.52	4.13
		DC	0	0	0	**100**	0	0	0	15.02	**0.03**	9.45			DC	100	0	0	0	0	0	0	37.50	30.80	15.09
		IS	0	0	0	80	16	3	1	16.34	2.47	13.43			IS	0	0	0	67	18	12	3	17.65	**1.25**	19.38
	0.7	HiTS	0	0	0	91	5	4	0	16.24	1.07	0.46		0.7	HiTS	0	0	0	**100**	0	0	0	16.81	**0.03**	0.71
		SBS	0	0	0	**100**	0	0	0	14.97	**0.00**	2.33			SBS	100	0	0	0	0	0	0	235.81	29.98	4.15
		DC	0	0	0	**99**	1	0	0	15.02	0.16	9.45			DC	100	0	0	0	0	0	0	278.98	45.20	14.94
		IS	0	0	0	88	10	2	0	15.75	1.57	13.41			IS	0	0	0	84	14	2	0	16.97	**0.59**	19.01
(M2)	0.01	HiTS	0	29	35	33	3	0	0	11.59	23.92	0.47	(M5)	0.01	HiTS	90	2	0	**8**	0	0	0	32.99	**8.34**	0.91
		SBS	0	31	51	18	0	0	0	10.50	27.28	2.32			SBS	100	0	0	0	0	0	0	11.00	25.40	5.77
		DC	0	28	29	**43**	0	0	0	11.59	**19.80**	9.46			DC	100	0	0	0	0	0	0	10.99	25.00	18.03
		IS	0	54	23	19	3	1	0	10.17	27.35	13.07			IS	100	0	0	0	0	0	0	11.38	24.96	20.34
	0.1	HiTS	0	0	3	89	5	3	0	17.38	4.51	0.46		0.1	HiTS	0	0	0	**100**	0	0	0	40.66	**0.18**	0.93
		SBS	0	0	0	**100**	0	0	0	15.12	**0.26**	2.33			SBS	100	0	0	0	0	0	0	86.24	25.00	5.85
		DC	0	0	2	98	0	0	0	15.35	1.15	9.44			DC	100	0	0	0	0	0	0	86.23	25.00	20.44
		IS	0	0	0	81	14	3	2	16.39	2.35	13.43			IS	0	0	0	41	20	21	18	41.53	**0.60**	29.45
	0.7	HiTS	0	0	0	92	4	4	0	17.16	1.28	0.46		0.7	HiTS	0	0	0	**100**	0	0	0	40.00	**0.00**	0.93
		SBS	0	0	0	**100**	0	0	0	14.97	**0.00**	2.34			SBS	100	0	0	0	0	0	0	591.67	25.00	5.95
		DC	0	0	0	**100**	0	0	0	14.97	**0.00**	9.46			DC	100	0	0	0	0	0	0	591.67	25.00	20.42
		IS	0	0	0	85	10	4	1	16.12	1.87	13.42			IS	0	0	0	**92**	7	1	0	40.07	**0.07**	29.29
(M3)	0.01	HiTS	0	7	53	39	1	0	0	9.14	16.43	0.59	(M6)	0.01	HiTS	82	14	2	**2**	0	0	0	4.44	**8.13**	1.31
		SBS	18	40	34	8	0	0	0	7.22	27.66	3.24			SBS	100	0	0	0	0	0	0	2.93	25.16	10.22
		DC	0	1	24	**75**	0	0	0	9.65	**7.32**	11.32			DC	100	0	0	0	0	0	0	4.36	**6.94**	30.72
		IS	4	10	30	47	6	3	0	9.33	13.33	15.32			IS	96	1	2	1	0	0	0	3.79	13.10	29.71
	0.1	HiTS	0	0	2	96	1	1	0	11.33	2.17	0.59		0.1	HiTS	0	7	1	**92**	0	0	0	6.60	**0.80**	1.32
		SBS	0	0	0	**100**	0	0	0	10.42	0.42	3.24			SBS	100	0	0	0	0	0	0	11.97	18.00	10.35
		DC	0	0	0	**100**	0	0	0	10.02	**0.02**	11.23			DC	99	1	0	0	0	0	0	7.49	5.10	30.76
		IS	0	0	0	83	12	4	1	10.64	1.56	15.49			IS	0	1	0	62	18	14	5	6.40	**1.15**	31.29
	0.7	HiTS	0	0	0	99	0	1	0	10.59	0.22	0.59		0.7	HiTS	0	0	0	**100**	0	0	0	6.34	**0.09**	1.31
		SBS	0	0	0	**100**	0	0	0	10.27	**0.04**	3.25			SBS	100	0	0	0	0	0	0	55.19	14.34	10.79
		DC	0	0	0	**100**	0	0	0	10.00	**0.00**	11.22			DC	100	0	0	0	0	0	0	27.24	5.10	34.30
		IS	0	0	0	91	7	1	1	10.34	0.86	15.49			IS	0	0	0	67	22	7	4	6.29	**0.88**	31.15

Table 5.8 Distribution of $\hat{N} - N$ for models (M1)-(M6) and all methods over 100 simulation runs in the "no-overlap" case with $n = 100$ in scenario (S1). Also the average MSE (Mean Squared Error) of the estimated signal $\hat{\boldsymbol{f}}$, the average Hausdorff distance d_H and the average computational time in seconds using an Intel Core i9 3.6 GHz CPU with 8 GB of RAM, all over 100 simulations. Bold: methods within 10% of the highest empirical frequency of $\hat{N} - N = 0$ or within 10% of the lowest empirical average $d_H(\times 10^2)$.

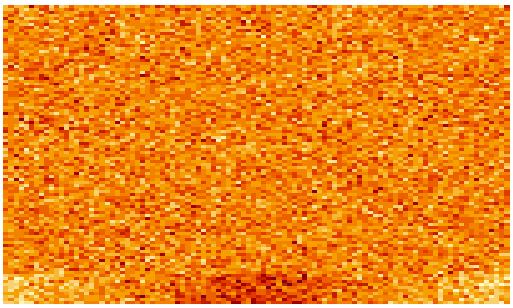
			$\hat{N} - N$												$\hat{N} - N$									
Model	sparsity	Method	≤-3	-2	-1	0	1	2	≥3	MSE	$d_H(\times 10^2)$	time	Model	sparsity	Method	≤-3	-2	-1	0	1	2	≥3	MSE	$d_H(\times 10^2)$	time
(M1)	0.01	HiTS	0	16	2	**77**	2	2	1	3.10	**10.29**	0.17	(M4)	0.01	HiTS	56	27	0	**17**	0	0	0	2.53	**22.79**	0.31
		SBS	0	99	1	0	0	0	0	1.89	33.99	1.19			SBS	100	0	0	0	0	0	0	1.34	50.00	1.53
		DC	0	65	0	35	0	0	0	2.28	22.55	2.89			DC	100	0	0	0	0	0	0	1.48	45.30	3.87
		IS	0	89	3	5	3	0	0	2.06	31.69	0.36			IS	100	0	0	0	0	0	0	1.38	48.52	0.75
	0.1	HiTS	0	0	0	93	3	3	1	3.56	2.41	0.16		0.1	HiTS	0	0	0	**99**	0	1	0	3.93	**1.27**	0.31
		SBS	0	0	27	73	0	0	0	4.20	11.37	1.21			SBS	100	0	0	0	0	0	0	6.02	29.21	1.54
		DC	0	0	0	**99**	1	0	0	3.07	**0.50**	2.85			DC	51	49	0	0	0	0	0	4.82	11.13	3.82
		IS	0	0	0	80	19	0	1	3.30	2.19	0.37			IS	0	1	0	65	29	3	2	3.68	**1.47**	0.78
	0.7	HiTS	0	0	0	92	2	4	2	3.37	0.88	0.15		0.7	HiTS	0	0	0	**99**	0	1	0	3.54	**0.16**	0.31
		SBS	0	0	10	90	0	0	0	10.91	6.17	1.21			SBS	100	0	0	0	0	0	0	34.44	20.81	1.55
		DC	0	0	0	**99**	1	0	0	2.99	**0.11**	2.87			DC	100	0	0	0	0	0	0	34.33	20.21	5.10
		IS	0	0	0	91	8	0	1	3.17	1.03	0.37			IS	0	0	0	84	12	4	0	3.47	**0.44**	0.78
(M2)	0.01	HiTS	0	62	5	**30**	3	0	0	2.07	**26.23**	0.17	(M5)	0.01	HiTS	100	0	0	0	0	0	0	4.69	**20.11**	0.38
		SBS	0	100	0	0	0	0	0	1.39	34.00	1.20			SBS	100	0	0	0	0	0	0	2.45	49.73	1.91
		DC	0	95	0	4	1	0	0	1.49	32.49	2.80			DC	100	0	0	0	0	0	0	2.45	49.75	5.11
		IS	0	96	4	0	0	0	0	1.44	33.94	0.36			IS	100	0	0	0	0	0	0	2.45	49.78	1.11
	0.1	HiTS	0	0	15	75	8	2	0	3.66	9.82	0.19		0.1	HiTS	0	2	0	**97**	0	1	0	8.64	**0.45**	0.38
		SBS	0	2	56	42	0	0	0	3.53	18.29	1.20			SBS	100	0	0	0	0	0	0	11.62	25.00	1.94
		DC	0	3	13	**83**	1	0	0	3.25	6.75	2.89			DC	100	0	0	0	0	0	0	11.62	24.95	5.30
		IS	0	0	0	**81**	18	0	1	3.39	**3.42**	0.38			IS	16	5	4	26	30	13	6	8.63	3.41	1.28
	0.7	HiTS	0	0	0	93	2	3	2	3.86	1.76	0.19		0.7	HiTS	0	0	0	**100**	0	0	0	8.00	**0.00**	0.38
		SBS	0	0	1	**99**	0	0	0	3.74	1.32	1.21			SBS	100	0	0	0	0	0	0	78.89	25.00	1.99
		DC	0	0	0	**99**	1	0	0	3.00	**0.12**	2.92			DC	100	0	0	0	0	0	0	78.89	25.00	7.02
		IS	0	0	0	90	9	0	1	3.23	1.22	0.38			IS	0	0	0	50	39	11	0	8.14	**0.36**	1.29
(M3)	0.01	HiTS	4	52	29	15	0	0	0	1.58	23.57	0.26	(M6)	0.01	HiTS	95	2	1	**2**	0	0	0	0.81	**14.75**	0.49
		SBS	70	27	3	0	0	0	0	1.21	42.59	1.35			SBS	100	0	0	0	0	0	0	0.62	44.91	2.78
		DC	0	40	32	**28**	0	0	0	1.64	**19.02**	3.25			DC	100	0	0	0	0	0	0	0.66	22.39	8.58
		IS	31	54	13	1	0	0	1	1.37	32.56	0.56			IS	100	0	0	0	0	0	0	0.60	30.60	1.95
	0.1	HiTS	0	2	37	61	0	0	0	2.32	11.60	0.26		0.1	HiTS	22	36	4	**38**	0	0	0	1.41	**3.47**	0.49
		SBS	0	28	50	22	0	0	0	2.62	20.05	1.35			SBS	100	0	0	0	0	0	0	1.70	19.66	2.81
		DC	0	0	10	**90**	0	0	0	2.12	3.49	3.24			DC	100	0	0	0	0	0	0	1.41	7.99	8.28
		IS	0	0	0	**96**	4	0	0	2.09	**0.70**	0.57			IS	7	30	8	33	12	7	3	1.33	**3.33**	2.00
	0.7	HiTS	0	0	0	97	2	1	0	2.38	0.96	0.27		0.7	HiTS	0	0	0	**100**	0	0	0	1.40	**0.16**	0.49
		SBS	0	0	2	98	0	0	0	4.43	2.64	1.37			SBS	100	0	0	0	0	0	0	7.55	17.17	2.92
		DC	0	0	0	**99**	1	0	0	2.03	**0.12**	3.28			DC	100	0	0	0	0	0	0	7.16	15.91	11.86
		IS	0	0	0	96	4	0	0	2.04	**0.26**	0.59			IS	0	0	0	79	15	6	0	1.26	**0.65**	2.01

Table 5.9 Distribution of $\hat{N} - N$ for models (M1)-(M6) and all methods over 100 simulation runs in the "no-overlap" case with $n = 300$ in scenario (S1). Also the average MSE (Mean Squared Error) of the estimated signal $\hat{\boldsymbol{f}}$, the average Hausdorff distance d_H and the average computational time in seconds using an Intel Core i9 3.6 GHz CPU with 8 GB of RAM, all over 100 simulations. Bold: methods within 10% of the highest empirical frequency of $\hat{N} - N = 0$ or within 10% of the lowest empirical average $d_H(\times 10^2)$.

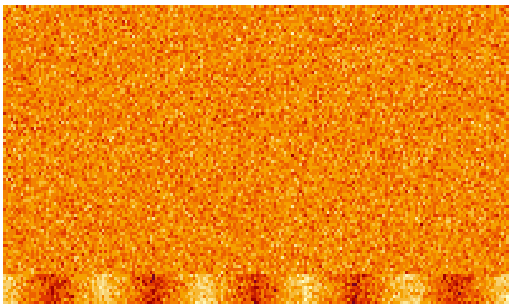
			$\hat{N} - N$												$\hat{N} - N$									
Model	sparsity	Method	≤-3	-2	-1	0	1	2	≥3	MSE	$d_H(\times 10^2)$	time	Model	sparsity	Method	≤-3	-2	-1	0	1	2	≥3	MSE	$d_H(\times 10^2)$	time
(M1)	0.01	HiTS	0	0	18	78	2	2	0	9.13	10.71	0.35	(M4)	0.01	HiTS	66	17	1	**16**	0	0	0	7.16	**18.30**	0.52
		SBS	0	14	55	31	0	0	0	7.53	22.30	1.77			SBS	100	0	0	0	0	0	0	3.18	45.64	2.88
		DC	0	0	10	**90**	0	0	0	8.85	**4.09**	6.20			DC	98	2	0	0	0	0	0	5.20	**20.29**	8.56
		IS	0	1	63	24	11	1	0	8.15	23.63	3.50			IS	100	0	0	0	0	0	0	3.63	24.86	5.14
	0.1	HiTS	0	0	0	95	1	4	0	9.95	1.49	0.35		0.1	HiTS	0	0	0	**100**	0	0	0	10.98	**0.77**	0.52
		SBS	0	0	28	72	0	0	0	12.02	10.64	1.77			SBS	100	0	0	0	0	0	0	16.79	24.36	2.90
		DC	0	0	0	**100**	0	0	0	9.06	**0.07**	6.18			DC	100	0	0	0	0	0	0	16.38	20.03	9.86
		IS	0	0	0	76	20	2	2	10.11	3.06	3.55			IS	0	0	0	74	15	7	4	10.63	**1.01**	5.61
	0.7	HiTS	0	0	0	95	0	5	0	9.69	**0.61**	0.35		0.7	HiTS	0	0	0	**100**	0	0	0	10.33	**0.06**	0.52
		SBS	0	0	18	82	0	0	0	40.72	7.87	1.77			SBS	100	0	0	0	0	0	0	102.88	20.24	2.93
		DC	0	0	0	**100**	0	0	0	9.01	**0.00**	6.18			DC	100	0	0	0	0	0	0	103.91	20.65	9.84
		IS	0	0	0	80	18	2	0	9.78	2.51	3.55			IS	0	0	0	80	11	5	4	10.45	**0.74**	5.61
(M2)	0.01	HiTS	0	22	62	15	1	0	0	6.50	28.69	0.36	(M5)	0.01	HiTS	100	0	0	0	0	0	0	14.03	**15.66**	0.65
		SBS	0	30	67	3	0	0	0	5.87	30.41	1.76			SBS	100	0	0	0	0	0	0	4.59	25.02	3.93
		DC	0	15	64	**21**	0	0	0	6.64	**26.31**	6.19			DC	100	0	0	0	0	0	0	4.59	25.00	11.43
		IS	0	41	47	10	1	1	0	6.01	30.73	3.46			IS	100	0	0	0	0	0	0	4.63	25.00	6.66
	0.1	HiTS	0	0	6	89	3	2	0	10.34	5.44	0.35		0.1	HiTS	0	0	0	**100**	0	0	0	24.74	**0.22**	0.67
		SBS	0	0	37	63	0	0	0	9.81	12.79	1.76			SBS	100	0	0	0	0	0	0	34.86	24.98	3.97
		DC	0	0	3	**97**	0	0	0	9.26	**1.76**	6.18			DC	100	0	0	0	0	0	0	34.85	25.00	13.55
		IS	0	0	0	74	21	4	1	10.21	3.46	3.55			IS	0	0	0	35	27	17	21	25.16	**0.60**	8.97
	0.7	HiTS	0	0	0	92	3	5	0	10.66	1.37	0.35		0.7	HiTS	0	0	0	**100**	0	0	0	24.14	**0.01**	0.66
		SBS	0	0	0	**100**	0	0	0	9.23	**0.12**	1.76			SBS	100	0	0	0	0	0	0	236.61	25.00	4.07
		DC	0	0	0	**100**	0	0	0	9.01	**0.00**	6.18			DC	100	0	0	0	0	0	0	236.62	25.00	13.53
		IS	0	0	0	79	19	2	0	9.81	2.62	3.55			IS	0	0	0	69	23	5	3	24.38	**0.20**	8.95
(M3)	0.01	HiTS	0	38	45	17	0	0	0	4.66	21.41	0.43	(M6)	0.01	HiTS	99	1	0	0	0	0	0	2.20	**11.05**	0.92
		SBS	2	43	46	9	0	0	0	4.45	23.64	2.33			SBS	100	0	0	0	0	0	0	1.38	23.87	6.31
		DC	0	17	41	**42**	0	0	0	5.13	**15.23**	7.15			DC	100	0	0	0	0	0	0	2.26	**11.27**	19.09
		IS	1	77	16	6	0	0	0	4.01	24.79	4.35			IS	100	0	0	0	0	0	0	1.56	21.38	10.44
	0.1	HiTS	0	0	16	82	1	1	0	6.70	5.92	0.43		0.1	HiTS	2	30	6	62	0	0	0	4.04	**2.03**	0.92
		SBS	0	3	29	68	0	0	0	7.17	9.51	2.33			SBS	100	0	0	0	0	0	0	4.76	18.75	6.42
		DC	0	0	0	**100**	0	0	0	6.14	**0.66**	7.16			DC	100	0	0	0	0	0	0	4.13	11.07	21.79
		IS	0	0	0	86	8	6	0	6.38	**1.38**	4.47			IS	0	2	1	**72**	17	6	2	3.80	**0.72**	11.01
	0.7	HiTS	0	0	0	96	3	1	0	6.89	0.76	0.43		0.7	HiTS	0	0	0	**100**	0	0	0	3.90	**0.11**	0.92
		SBS	0	0	0	**100**	0	0	0	8.14	0.60	2.33			SBS	100	0	0	0	0	0	0	18.86	15.05	6.72
		DC	0	0	0	**100**	0	0	0	6.01	**0.00**	7.17			DC	100	0	0	0	0	0	0	26.47	20.50	23.10
		IS	0	0	0	85	11	4	0	6.34	1.41	4.47			IS	0	0	0	82	12	5	1	3.71	**0.48**	11.03

Table 5.10 Distribution of $\hat{N} - N$ for models (M1)-(M6) and all methods over 100 simulation runs in the "no-overlap" case with $n = 500$ in scenario (S1). Also the average MSE (Mean Squared Error) of the estimated signal $\hat{\boldsymbol{f}}$, the average Hausdorff distance d_H and the average computational time in seconds using an Intel Core i9 3.6 GHz CPU with 8 GB of RAM, all over 100 simulations. Bold: methods within 10% of the highest empirical frequency of $\hat{N} - N = 0$ or within 10% of the lowest empirical average $d_H(\times 10^2)$.

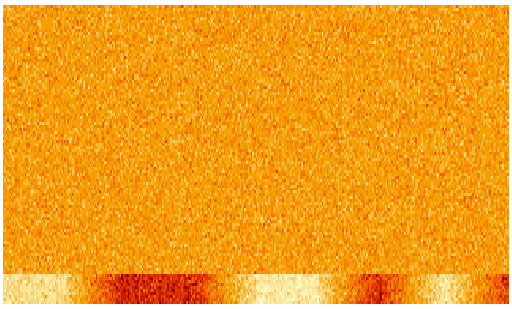
			$\hat{N} - N$												$\hat{N} - N$									
Model	sparsity	Method	≤-3	-2	-1	0	1	2	≥3	MSE	$d_H(\times 10^2)$	time	Model	sparsity	Method	≤-3	-2	-1	0	1	2	≥3	MSE	$d_H(\times 10^2)$	time
(M1)	0.01	HiTS	0	0	13	79	5	3	0	15.57	8.59	0.46	(M4)	0.01	HiTS	30	25	4	**41**	0	0	0	14.65	**10.04**	0.72
		SBS	0	2	56	42	0	0	0	13.24	19.19	2.33			SBS	100	0	0	0	0	0	0	5.61	40.20	4.08
		DC	0	0	4	**95**	1	0	0	14.97	**2.34**	9.54			DC	98	2	0	0	0	0	0	8.92	18.20	13.46
		IS	0	0	23	65	6	4	2	15.45	10.73	13.37			IS	100	0	0	0	0	0	0	6.02	20.42	16.87
	0.1	HiTS	0	0	0	90	6	3	1	16.82	1.83	0.46		0.1	HiTS	0	0	0	**100**	0	0	0	17.98	**0.73**	0.73
		SBS	0	0	19	81	0	0	0	18.89	7.82	2.34			SBS	100	0	0	0	0	0	0	27.63	22.15	4.08
		DC	0	0	0	**99**	1	0	0	15.06	**0.20**	9.47			DC	100	0	0	0	0	0	0	27.35	20.01	15.20
		IS	0	0	0	81	14	3	2	16.42	2.37	13.44			IS	0	0	0	64	19	13	4	17.75	**1.31**	19.45
	0.7	HiTS	0	0	0	90	7	3	0	16.12	1.03	0.46		0.7	HiTS	0	0	0	**100**	0	0	0	16.83	**0.03**	0.75
		SBS	0	0	19	81	0	0	0	70.65	8.30	2.34			SBS	100	0	0	0	0	0	0	171.52	20.15	4.29
		DC	0	0	0	**100**	0	0	0	14.97	**0.00**	9.54			DC	100	0	0	0	0	0	0	174.04	20.91	15.52
		IS	0	0	0	86	11	3	0	15.92	1.83	13.42			IS	0	0	0	81	15	4	0	17.07	**0.72**	19.86
(M2)	0.01	HiTS	0	11	66	21	2	0	0	11.61	26.66	0.47	(M5)	0.01	HiTS	94	4	0	**2**	0	0	0	29.64	**9.99**	0.96
		SBS	0	8	82	10	0	0	0	10.90	27.96	2.33			SBS	100	0	0	0	0	0	0	7.57	25.06	6.00
		DC	0	8	60	**31**	1	0	0	11.88	**23.10**	9.47			DC	100	0	0	0	0	0	0	7.57	24.97	18.83
		IS	0	23	60	16	0	1	0	10.81	28.56	13.14			IS	100	0	0	0	0	0	0	7.65	24.90	21.37
	0.1	HiTS	0	0	5	87	5	3	0	17.17	5.27	0.47		0.1	HiTS	0	0	0	**100**	0	0	0	40.50	**0.15**	0.95
		SBS	0	0	17	83	0	0	0	15.69	6.22	2.34			SBS	100	0	0	0	0	0	0	58.08	24.97	5.84
		DC	0	0	3	**96**	1	0	0	15.30	**1.66**	9.49			DC	100	0	0	0	0	0	0	58.06	25.00	20.43
		IS	0	0	0	80	14	4	2	16.60	2.69	13.45			IS	0	0	0	22	26	20	32	42.13	**0.75**	29.64
	0.7	HiTS	0	0	0	91	5	4	0	17.58	1.34	0.47		0.7	HiTS	0	0	0	**100**	0	0	0	39.97	**0.00**	0.96
		SBS	0	0	0	**100**	0	0	0	15.16	**0.06**	2.34			SBS	100	0	0	0	0	0	0	394.27	25.00	5.96
		DC	0	0	0	**100**	0	0	0	14.97	**0.00**	9.55			DC	100	0	0	0	0	0	0	394.29	25.00	20.43
		IS	0	0	0	83	14	3	0	16.10	2.16	13.44			IS	0	0	0	71	26	2	1	40.35	**0.20**	29.39
(M3)	0.01	HiTS	0	35	47	17	1	0	0	7.66	21.06	0.60	(M6)	0.01	HiTS	94	6	0	0	0	0	0	3.76	**11.83**	1.35
		SBS	1	76	22	1	0	0	0	6.27	25.10	3.25			SBS	100	0	0	0	0	0	0	2.48	21.40	10.34
		DC	0	24	41	**35**	0	0	0	8.08	**16.71**	11.26			DC	100	0	0	0	0	0	0	3.56	**12.65**	30.73
		IS	0	85	11	4	0	0	0	6.09	24.50	15.04			IS	100	0	0	0	0	0	0	2.62	21.34	29.24
	0.1	HiTS	0	0	8	90	2	0	0	11.09	3.94	0.60		0.1	HiTS	1	15	2	**82**	0	0	0	6.59	**1.27**	1.35
		SBS	0	0	20	79	1	0	0	11.57	6.64	3.25			SBS	100	0	0	0	0	0	0	7.92	18.88	10.43
		DC	0	0	0	**100**	0	0	0	10.07	**0.19**	11.25			DC	100	0	0	0	0	0	0	6.95	14.01	34.40
		IS	0	0	0	83	12	4	1	10.65	1.58	15.52			IS	0	1	0	59	21	13	6	6.45	**1.18**	31.40
	0.7	HiTS	0	0	0	**99**	0	1	0	10.73	0.38	0.60		0.7	HiTS	0	0	0	**100**	0	0	0	6.37	**0.10**	1.35
		SBS	0	0	0	**100**	0	0	0	11.21	0.18	3.26			SBS	100	0	0	0	0	0	0	29.65	13.99	10.86
		DC	0	0	0	**100**	0	0	0	10.00	**0.00**	11.25			DC	100	0	0	0	0	0	0	45.66	21.23	34.73
		IS	0	0	0	87	10	2	1	10.48	1.20	15.52			IS	0	0	0	64	23	9	4	6.32	**0.90**	31.46



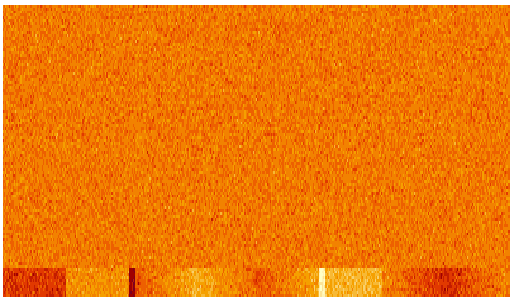
(a) (M1) one



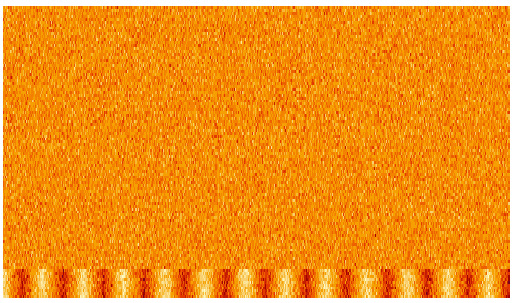
(b) (M2) wave



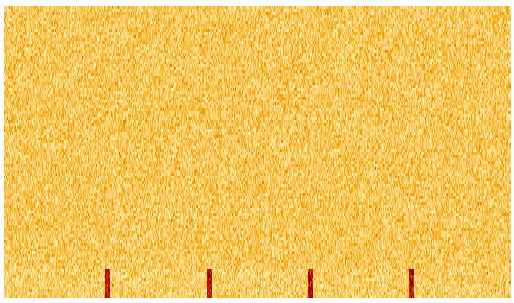
(c) (M3) mix1



(d) (M4) mix2



(e) (M5) extreme.wave



(f) (M6) lin.sgmts

Fig. 5.5 Examples of data with its underlying signal studied in Section 5.4.2 in scenario (S2). (a)-(f) visualisation of the data matrix $\boldsymbol{X}$ when n=100, sparsity=0.1 and "complete-overlap" case.

Table 5.11 Distribution of $\hat{N} - N$ for models (M1)-(M6) and all methods over 100 simulation runs in the "complete-overlap" case in scenario (S2). Also the average MSE (Mean Squared Error) of the estimated signal $\hat{\boldsymbol{f}}$, the average Hausdorff distance d_H and the average computational time in seconds using an Intel Core i9 3.6 GHz CPU with 8 GB of RAM, all over 100 simulations.

			$\hat{N}-N$												$\hat{N}-N$									
Model	n	sparsity	≤-3	-2	-1	0	1	2	≥3	MSE	$d_H(\times 10^2)$	time	Model	n	sparsity	≤-3	-2	-1	0	1	2	≥3	MSE	$d_H(\times 10^2)$	time
		0.01	0	0	6	91	2	1	0	4.07	11.24	2.49			0.01	6	27	33	28	4	2	0	4.93	8.13	8.20
	100	0.1	0	0	0	97	2	1	0	4.33	6.63	2.49		100	0.1	0	0	4	91	5	0	0	5.84	3.59	8.01
		0.7	0	0	0	93	5	2	0	5.88	7.78	2.52			0.7	0	0	0	92	8	0	0	9.38	2.72	10.19
		0.01	0	0	0	96	3	1	0	12.44	8.39	6.88			0.01	1	8	29	47	15	0	0	15.29	6.10	22.61
(M1)	300	0.1	0	0	0	97	2	1	0	12.79	5.73	6.93	(M4)	300	0.1	0	0	0	94	6	0	0	16.91	2.92	22.31
		0.7	0	0	0	92	6	2	0	18.55	8.27	7.11			0.7	0	0	1	85	14	0	0	30.90	3.50	30.15
		0.01	0	0	0	96	4	0	0	20.62	9.36	11.22			0.01	0	1	22	56	20	1	0	25.95	5.26	37.79
	500	0.1	0	0	0	95	3	2	0	22.59	9.43	11.30		500	0.1	0	0	0	97	3	0	0	28.08	3.11	38.57
		0.7	0	0	0	98	1	1	0	28.37	7.16	11.28			0.7	0	0	8	88	4	0	0	59.36	4.58	72.39
		0.01	21	20	29	30	0	0	0	8.99	7.68	4.46			0.01	2	6	25	67	0	0	0	10.23	1.87	9.26
	100	0.1	0	0	0	97	3	0	0	10.99	2.77	4.41		100	0.1	0	0	0	100	0	0	0	11.51	1.15	9.46
		0.7	0	0	0	96	4	0	0	15.60	2.62	4.77			0.7	0	0	0	100	0	0	0	19.80	1.24	11.28
		0.01	1	6	20	73	0	0	0	29.71	4.24	12.27			0.01	0	0	2	98	0	0	0	30.68	1.42	25.96
(M2)	300	0.1	0	0	0	100	0	0	0	32.66	2.50	12.48	(M5)	300	0.1	0	0	1	99	0	0	0	34.14	1.14	27.53
		0.7	0	0	0	96	4	0	0	49.78	2.70	15.51			0.7	0	0	2	93	5	0	0	67.19	1.52	39.85
		0.01	0	4	16	80	0	0	0	49.76	3.95	19.81			0.01	0	0	3	96	1	0	0	50.94	1.35	42.24
	500	0.1	0	0	0	99	1	0	0	54.93	2.90	20.41		500	0.1	0	0	2	98	0	0	0	56.68	1.22	46.36
		0.7	0	0	0	97	3	0	0	82.23	2.71	25.50			0.7	0	1	7	85	6	1	0	124.03	1.66	75.44
		0.01	3	6	54	37	0	0	0	4.88	7.26	6.78			0.01	0	4	1	82	12	1	0	3.86	2.59	10.54
	100	0.1	0	1	20	76	3	0	0	6.93	5.81	7.32		100	0.1	0	0	0	99	1	0	0	3.67	0.24	9.11
		0.7	0	0	7	75	17	1	0	16.18	5.24	8.31			0.7	0	0	0	100	0	0	0	3.60	0.20	8.83
		0.01	0	1	39	59	1	0	0	15.04	6.43	19.17			0.01	0	1	0	91	8	0	0	11.10	0.92	28.09
(M3)	300	0.1	0	0	10	88	2	0	0	19.53	5.08	20.50	(M6)	300	0.1	0	0	0	99	1	0	0	10.84	0.26	26.00
		0.7	0	0	0	90	10	0	0	43.21	4.77	30.85			0.7	0	0	0	100	0	0	0	10.81	0.20	25.34
		0.01	0	0	41	59	0	0	0	24.90	6.36	31.51			0.01	0	0	0	97	3	0	0	18.25	0.52	46.03
	500	0.1	0	0	11	87	2	0	0	32.69	5.38	33.49		500	0.1	0	0	0	100	0	0	0	17.98	0.20	41.82
		0.7	0	0	2	64	33	1	0	79.18	5.06	63.15			0.7	0	0	0	100	0	0	0	17.98	0.20	41.89

Table 5.12 Distribution of $\hat{N} - N$ for models (M1)-(M6) and all methods over 100 simulation runs in the "half-overlap" case in scenario (S2). Also the average MSE (Mean Squared Error) of the estimated signal $\hat{\boldsymbol{f}}$, the average Hausdorff distance d_H and the average computational time in seconds using an Intel Core i9 3.6 GHz CPU with 8 GB of RAM, all over 100 simulations.

			$\hat{N}-N$												$\hat{N}-N$									
Model	n	sparsity	≤-3	-2	-1	0	1	2	≥3	MSE	$d_H(\times 10^2)$	time	Model	n	sparsity	≤-3	-2	-1	0	1	2	≥3	MSE	$d_H(\times 10^2)$	time
		0.01	0	0	6	91	2	1	0	4.07	11.24	2.53			0.01	6	27	33	28	4	2	0	4.93	8.13	8.27
	100	0.1	0	0	0	97	2	1	0	4.33	6.63	2.54		100	0.1	0	0	13	82	4	1	0	5.59	4.25	8.02
		0.7	0	0	0	93	5	2	0	5.88	7.78	2.56			0.7	0	0	1	93	6	0	0	8.81	3.51	9.79
		0.01	0	0	0	96	3	1	0	12.44	8.39	7.18			0.01	5	29	33	26	7	0	0	14.10	8.40	23.11
(M1)	300	0.1	0	0	0	97	2	1	0	12.79	5.73	7.08	(M4)	300	0.1	0	1	2	94	3	0	0	16.34	3.31	22.57
		0.7	0	0	0	92	6	2	0	18.55	8.27	7.27			0.7	0	2	6	75	17	0	0	38.58	5.19	33.69
		0.01	0	0	0	96	4	0	0	20.62	9.36	11.41			0.01	0	12	27	46	15	0	0	25.03	6.75	39.38
	500	0.1	0	0	0	95	3	2	0	22.59	9.43	11.51		500	0.1	0	0	5	89	6	0	0	27.24	3.92	39.39
		0.7	0	0	0	98	1	1	0	28.37	7.16	11.46			0.7	0	0	18	66	16	0	0	53.20	6.07	63.15
		0.01	21	20	29	30	0	0	0	8.99	7.68	4.54			0.01	2	6	25	67	0	0	0	10.23	1.87	9.42
	100	0.1	0	1	9	90	0	0	0	10.83	3.56	4.57		100	0.1	0	0	12	87	1	0	0	11.52	1.65	10.43
		0.7	0	0	0	93	7	0	0	15.52	3.03	4.98			0.7	0	0	5	80	15	0	0	21.20	1.77	13.38
		0.01	13	24	36	27	0	0	0	26.78	6.91	12.64			0.01	7	15	21	57	0	0	0	29.77	2.03	26.78
(M2)	300	0.1	0	0	2	98	0	0	0	32.43	3.09	12.82	(M5)	300	0.1	0	0	0	98	2	0	0	33.78	1.34	29.58
		0.7	0	0	0	97	3	0	0	46.90	3.04	15.02			0.7	0	0	8	87	5	0	0	66.94	1.71	41.26
		0.01	2	11	22	65	0	0	0	48.33	4.41	20.21			0.01	0	1	12	87	0	0	0	50.66	1.60	43.96
	500	0.1	0	0	0	100	0	0	0	54.38	3.21	20.92		500	0.1	0	0	4	94	2	0	0	56.34	1.49	50.29
		0.7	0	0	0	98	2	0	0	74.22	2.90	24.64			0.7	0	0	8	84	8	0	0	115.23	1.79	78.92
		0.01	3	6	54	37	0	0	0	4.88	7.26	6.80			0.01	0	4	1	82	12	1	0	3.86	2.59	10.65
	100	0.1	0	0	0	95	5	0	0	6.74	5.87	7.22		100	0.1	0	0	0	99	1	0	0	3.70	0.34	9.39
		0.7	0	0	0	86	14	0	0	13.83	5.20	8.97			0.7	0	0	0	100	0	0	0	3.60	0.20	9.32
		0.01	0	1	10	86	3	0	0	15.47	6.34	19.09			0.01	0	2	0	85	12	1	0	11.14	1.68	30.71
(M3)	300	0.1	0	0	0	91	9	0	0	19.28	5.38	20.66	(M6)	300	0.1	0	0	0	100	0	0	0	10.85	0.22	28.22
		0.7	0	0	0	88	12	0	0	38.65	5.13	33.13			0.7	0	0	0	100	0	0	0	10.94	0.22	26.59
		0.01	0	0	6	90	4	0	0	25.96	6.27	31.56			0.01	0	2	0	93	5	0	0	18.29	1.05	47.69
	500	0.1	0	0	0	92	8	0	0	31.63	5.44	34.03		500	0.1	0	0	0	99	1	0	0	18.01	0.21	43.01
		0.7	0	0	0	67	28	5	0	69.38	5.11	65.61			0.7	0	0	0	100	0	0	0	18.06	0.20	43.91

Table 5.13 Distribution of $\hat{N} - N$ for models (M1)-(M6) and all methods over 100 simulation runs in the "no-overlap" case in scenario (S2). Also the average MSE (Mean Squared Error) of the estimated signal $\hat{\boldsymbol{f}}$, the average Hausdorff distance d_H and the average computational time in seconds using an Intel Core i9 3.6 GHz CPU with 8 GB of RAM, all over 100 simulations.

			$\hat{N} - N$												$\hat{N} - N$									
Model	n	sparsity	≤-3	-2	-1	0	1	2	≥3	MSE	$d_H(\times 10^2)$	time	Model	n	sparsity	≤-3	-2	-1	0	1	2	≥3	MSE	$d_H(\times 10^2)$	time
		0.01	0	0	6	91	2	1	0	4.07	11.24	2.54			0.01	6	27	33	28	4	2	0	4.93	8.13	8.26
	100	0.1	0	0	0	97	2	1	0	4.33	6.63	2.54		100	0.1	1	13	34	44	7	1	0	5.68	7.22	8.38
		0.7	0	0	0	93	5	2	0	5.88	7.78	2.56			0.7	0	0	40	59	1	0	0	8.34	7.26	8.57
		0.01	0	0	0	96	3	1	0	12.44	8.39	7.04			0.01	33	32	20	15	0	0	0	12.80	10.29	23.75
(M1)	300	0.1	0	0	0	97	2	1	0	12.79	5.73	7.09	(M4)	300	0.1	0	0	26	70	4	0	0	16.17	5.57	22.91
		0.7	0	0	0	92	6	2	0	18.55	8.27	7.27			0.7	0	3	30	64	3	0	0	28.78	6.55	29.80
		0.01	0	0	0	96	4	0	0	20.62	9.36	11.43			0.01	17	28	41	14	0	0	0	22.30	10.39	41.47
	500	0.1	0	0	0	95	3	2	0	22.59	9.43	11.51		500	0.1	0	4	25	64	7	0	0	27.14	5.66	41.25
		0.7	0	0	0	98	1	1	0	28.37	7.16	11.50			0.7	0	0	11	81	8	0	0	43.61	6.25	46.16
		0.01	21	20	29	30	0	0	0	8.99	7.68	4.54			0.01	2	6	25	67	0	0	0	10.23	1.87	9.35
	100	0.1	5	29	56	10	0	0	0	10.41	7.20	5.53		100	0.1	17	28	45	10	0	0	0	11.84	2.83	16.85
		0.7	0	0	39	60	1	0	0	15.97	5.58	6.48			0.7	2	8	39	47	4	0	0	25.33	2.68	28.50
		0.01	57	33	10	0	0	0	0	22.39	10.63	14.42			0.01	79	19	2	0	0	0	0	26.46	3.27	41.24
(M2)	300	0.1	1	16	65	18	0	0	0	31.35	6.89	16.11	(M5)	300	0.1	19	27	40	14	0	0	0	35.41	2.85	73.54
		0.7	0	0	39	57	4	0	0	44.93	5.62	18.91			0.7	4	12	29	44	9	2	0	76.21	2.68	113.79
		0.01	46	42	11	1	0	0	0	39.17	9.09	23.82			0.01	64	18	17	1	0	0	0	45.17	3.09	78.29
	500	0.1	1	12	62	25	0	0	0	52.41	6.89	27.55		500	0.1	10	21	46	23	0	0	0	59.01	2.72	118.72
		0.7	0	0	48	49	3	0	0	75.02	6.13	32.26			0.7	3	23	37	30	6	1	0	127.69	2.91	162.68
		0.01	3	6	54	37	0	0	0	4.88	7.26	6.89			0.01	0	4	1	82	12	1	0	3.86	2.59	10.70
	100	0.1	0	0	3	90	7	0	0	6.16	5.90	7.08		100	0.1	0	0	0	100	0	0	0	3.73	0.55	9.33
		0.7	0	0	0	78	21	1	0	10.71	5.51	8.90			0.7	0	0	0	100	0	0	0	3.60	0.20	9.05
		0.01	0	1	29	69	1	0	0	15.04	6.60	19.28			0.01	1	2	9	74	13	1	0	11.06	2.21	30.91
(M3)	300	0.1	0	0	0	87	13	0	0	17.64	5.05	20.73	(M6)	300	0.1	0	0	0	99	1	0	0	10.94	0.31	27.31
		0.7	0	0	2	83	15	0	0	32.96	5.77	32.25			0.7	0	0	0	100	0	0	0	10.97	0.22	26.93
		0.01	0	3	29	66	2	0	0	24.69	6.50	32.37			0.01	0	0	19	72	9	0	0	18.06	1.33	45.45
	500	0.1	0	0	0	89	11	0	0	29.30	5.17	36.79		500	0.1	0	0	0	99	1	0	0	18.08	0.22	40.25
		0.7	0	0	1	58	38	3	0	59.59	5.75	66.94			0.7	0	0	0	100	0	0	0	17.98	0.20	40.61

5.5 Data applications

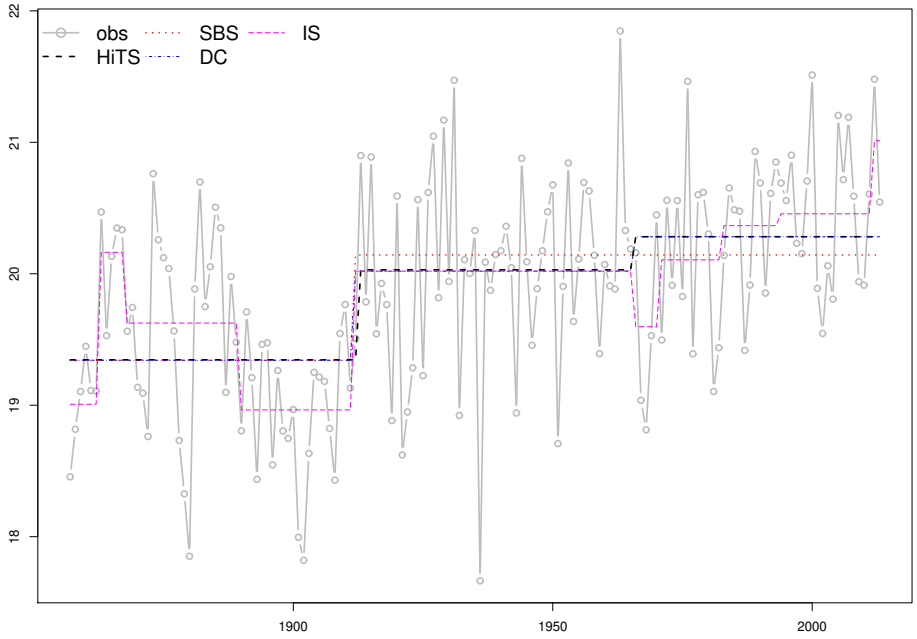
5.5.1 Average January temperatures in South Africa

We study a land temperature dataset available from http://berkeleyearth.org. As introduced earlier in Section 5.1, this data set consists of average temperatures in January recorded in 50 cities of South Africa from 1857 to 2013. The curves of 50 cities are shown in Figure 5.1 and they appear to have cross-sectional dependence. As studied in Section 5.3, assuming cross-sectional correlation does not affect the consistency results, thus we use the threshold that is the same as one used in simulations.

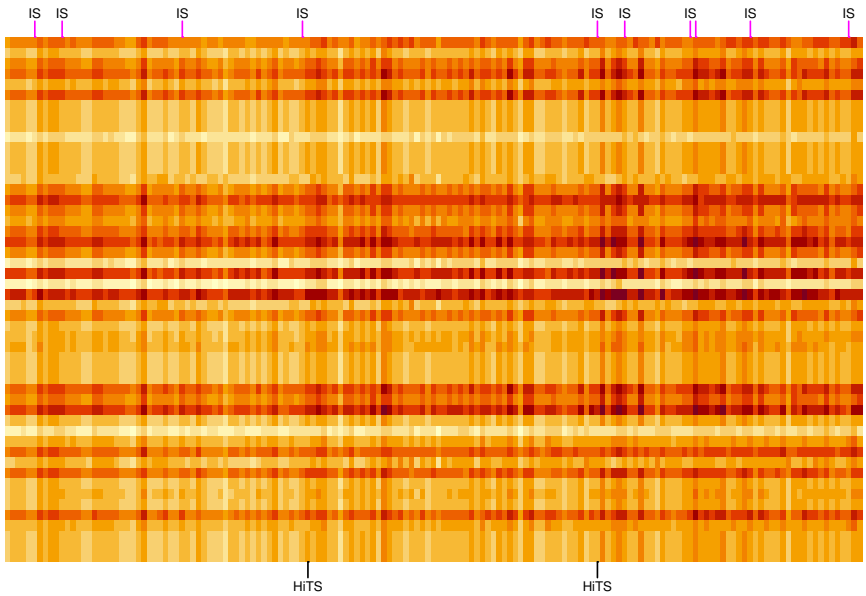
The HiTS algorithm identifies 2 change-points, 1912 and 1965, while SBS detects a change in 1911 and DC returns 2 change-points, 1911 and 1965. IS reports 9 change-points that include 1911 and 1965. Figure 5.6a shows that HiTS and DC share one change-point in 1965 and that SBS and DC share one point in 1911 where the estimated signals from all methods except IS return positive mean changes. Figure 5.6b shows the time location of 9 change-points detected by IS where two of them are either very close or equal to the estimated change-points by HiTS.

As an effort to find which coordinate(s) is associated with each of those estimated change-point, using $\tilde{\boldsymbol{f}}$ as an input data matrix, we first compute the (non-aggregated) detail coefficients for each change-point returned by HiTS as given in (5.15). Then we perform hard thresholding on those detail coefficients where the threshold is the same as the one used Section 5.2.4. We refer to this process as "post-thresholding".

Specifically, in the post-thresholding, two estimated change-points in 1912 and 1965 for all 50 cities are treated separately, thus each of 100 estimated change-points should be either survived or removed. For example, if the detail coefficient computed for the change of Cape town in 1912 is survived from the post-thresholding then we consider the change in 1912 is relevant to Cape town otherwise we conclude the



(a) Cape town



(b) 50 cities in South Africa

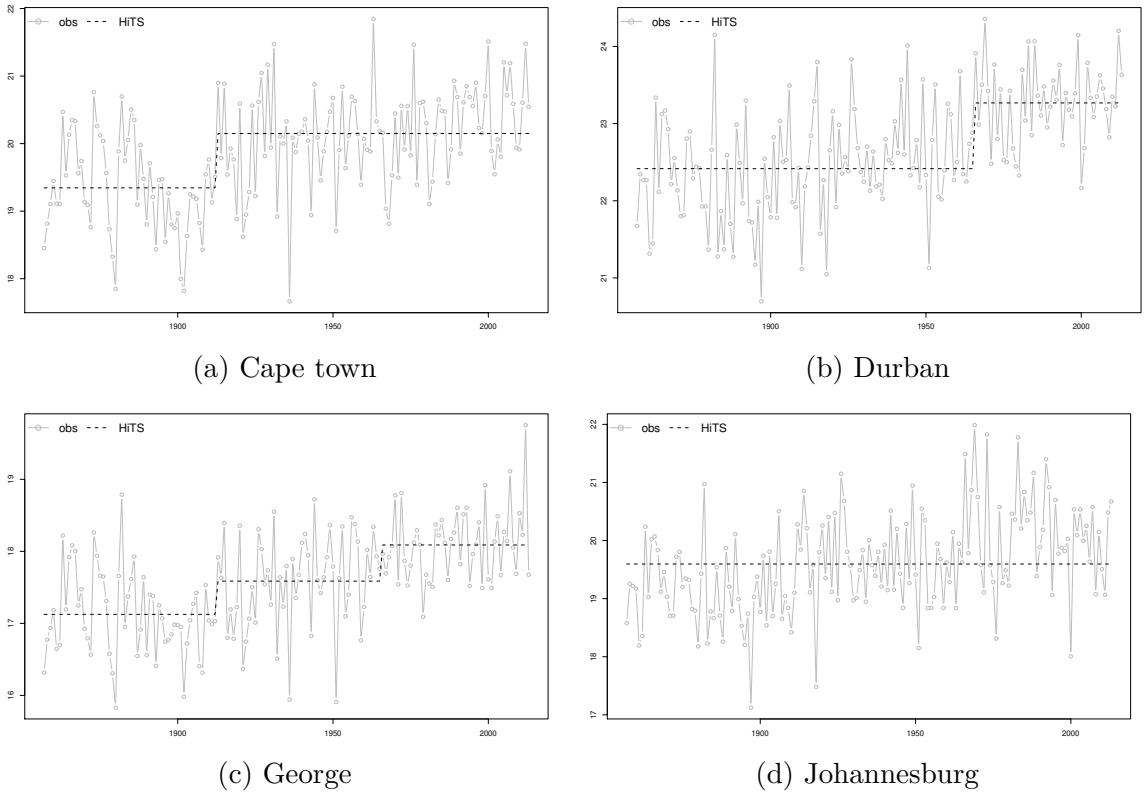
Fig. 5.6 Change-point analysis for January average temperature curves of 50 cities in South Africa from 1857 to 2013 in Section 5.5.1. (a) the data series (grey dots) of Cape town and estimated signal with change-points returned by HiTS (- - -), SBS (·····), DC (-·-·-) and IS(----), (b) visualisation of the data matrix and estimated change-points returned by HiTS and IS.

Table 5.14 Fifty cities in South Africa classified into four categories by the post-thresholding of the HiTS algorithm described in Section 5.5.1.

Estimated change-points	Cities
1912, 1965	George
1912	Cape Town, Paarl, Somerset West, Worcester
1965	Bisho, Durban, East London, Port Elizabeth, Richards Bay, Uitenhage
None	Alberton, Benoni, Bethal, Bloemfontein, Boksburg, Botshabelo, Brakpan, Brits, Johannesburg, Kimberley, Klerksdorp, Kroonstad, Krugersdorp, Middelburg, Midrand, Nelspruit, Newcastle, Nigel, Orkney, Phalaborwa, Pietermaritzburg, Pietersburg, Potchefstroom, Potgietersrus, Pretoria, Queenstown, Randfontein, Rustenburg, Soweto, Springs, Tembisa, Vanderbijlpark, Vereeniging, Verwoerdburg, Virginia, Vryheid, Welkom, Westonaria, Witbank

change is not associated with the corresponding city. As shown in Table 5.14, after the post-thresholding, the curves of 50 cities can be classified into four categories: cities including 1) two change-points in 1912 and 1965, 2) only one change-point in 1912, 3) only one change-point in 1965 and 4) no change-points. Figure 5.7 shows one randomly selected city from each category.

To see whether this classification includes any useful information, we mark the location of each city on a South Africa map as shown in Figure 5.8. Interestingly, those cities from the same category are geographically close to each other. Especially, George is the only city in which both estimated change-points in 1912 and 1965 are survived and Figure 5.8 shows that George is geographically located between two groups of cities, one including the estimated change-point only in 1912 and the other having only in 1965. This example shows the possibility that the HiTS procedure can be a useful first step for a higher-level representation of the high-dimensional panel data e.g. time series classification.



(a) Cape town (b) Durban

(c) George (d) Johannesburg

Fig. 5.7 The post-thresholded HiTS estimate of the piecewise-constant trend for January average temperature curves of 4 cities in South Africa from 1857 to 2013 in Section 5.5.1. (a) the data series (grey dots); the HiTS estimate (- - -) for average temperature of Cape town, (b) Durban, (c) George and (d) Johannesburg.

5.5.2 Monthly average sea ice extent of Arctic and Antarctic

We analyse the sea ice extent of the Arctic and the Antarctic available from https://nsidc.org to estimate the change-points in its linear trend. This dataset consists of the monthly average sea ice extent of the Arctic and the Antarctic from 1979 to 2018 where the curves of all months are shown in Figures 5.9 and 5.11, respectively.

Figure 5.9 shows the well-known decreasing trend of the average sea ice extent in the Arctic and the HiTS estimate identifies change-points in 1988, 2003 and 2007. This is not much different from the estimated change-points obtained in Section 4.5.2 where February and September are analysed separately as a univariate data sequence and

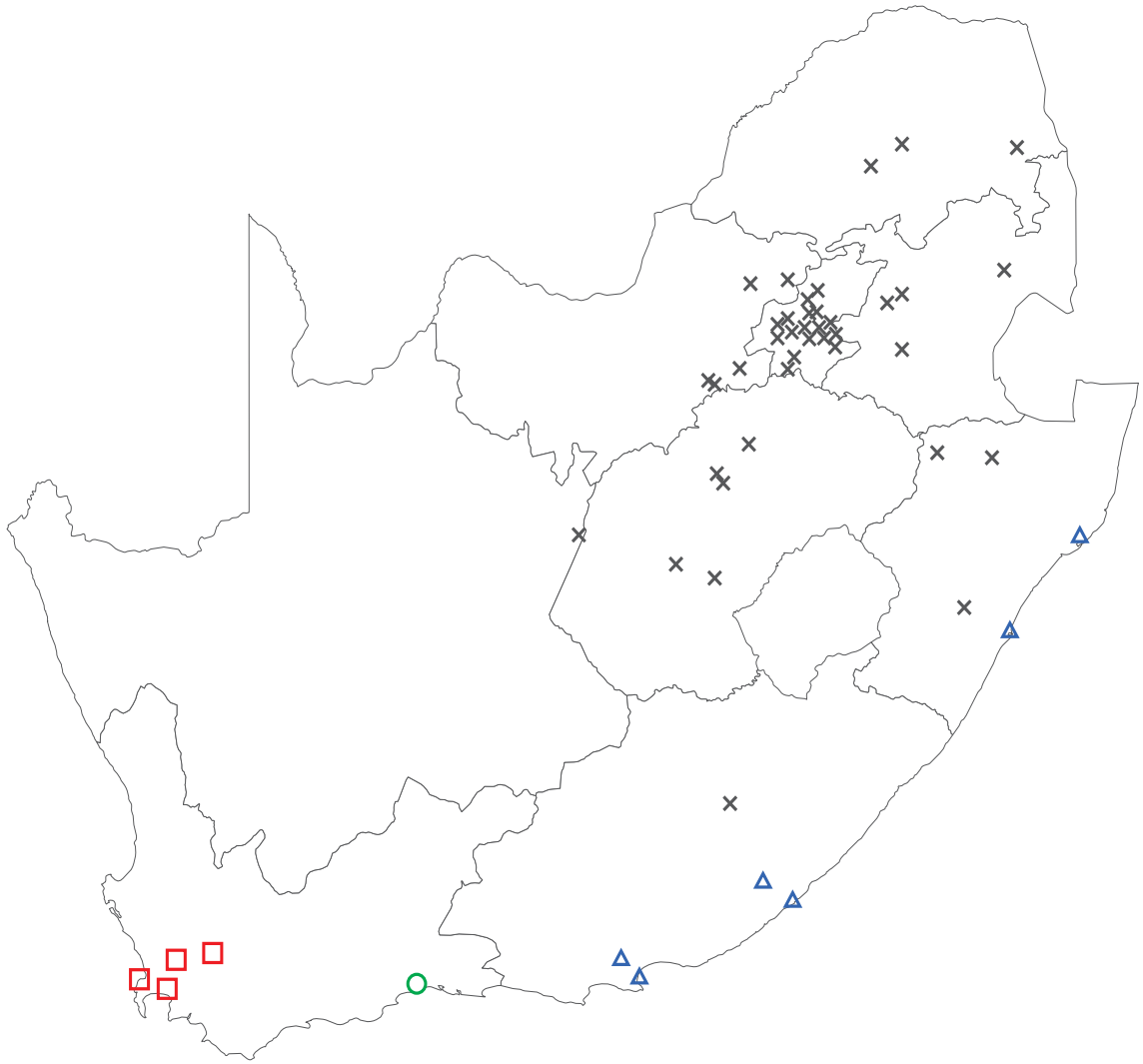


Fig. 5.8 The geographical locations of 50 cities in South Africa that are classified into four categories by the post-thresholding of the HiTS algorithm described in Section 5.5.1; cities with estimated change-points in 1912 and 1965 (◯), in 1912 (□), in 1965 (△) and those with no estimated change-points (×).

two change-points in 2004 and 2007 are identified in February and one change-point in 2006 is detected in September.

As done in Section 5.5.1, to examine which month is associated with each of those three change-points, we perform post-thresholding on the estimated functions for 12 months. Figure 5.10 indicates that two change-points in 2003 and 2007 are survived in

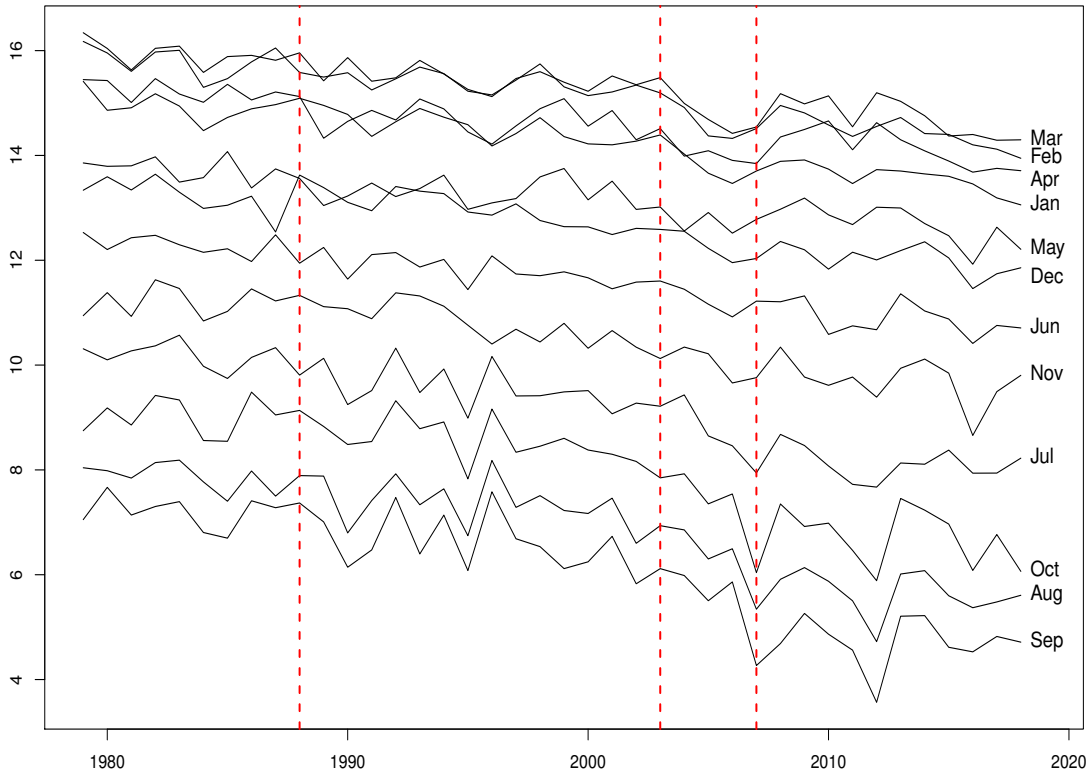


Fig. 5.9 The monthly average sea ice extent in the Arctic from 1979 to 2018 analysed in Section 5.5.2 (——) and the estimated change-points returned by HiTS (┆).

January and February while the change-point in 1988 is survived only in December. All three change-points are thresholded in other months from March to November.

Unlike the gentle decreasing trend shown in the sea ice extent of the Arctic, Figure 5.11 shows that the sea ice extent of the Antarctic has a modest increasing trend until recent years. However, at the same time, relatively strong decreasing trends are observed in most of the months from around 2016 and this is identified by the HiTS estimate which detects change-points in 1980, 1983 and 2015.

As in the Arctic data, we perform post-thresholding on the curves of 12 months. Figure 5.12 shows that the estimated change-point in 2015 is survived in all months except January while the change-point in 1980 survives only in April and the one in 1983 does so in October.

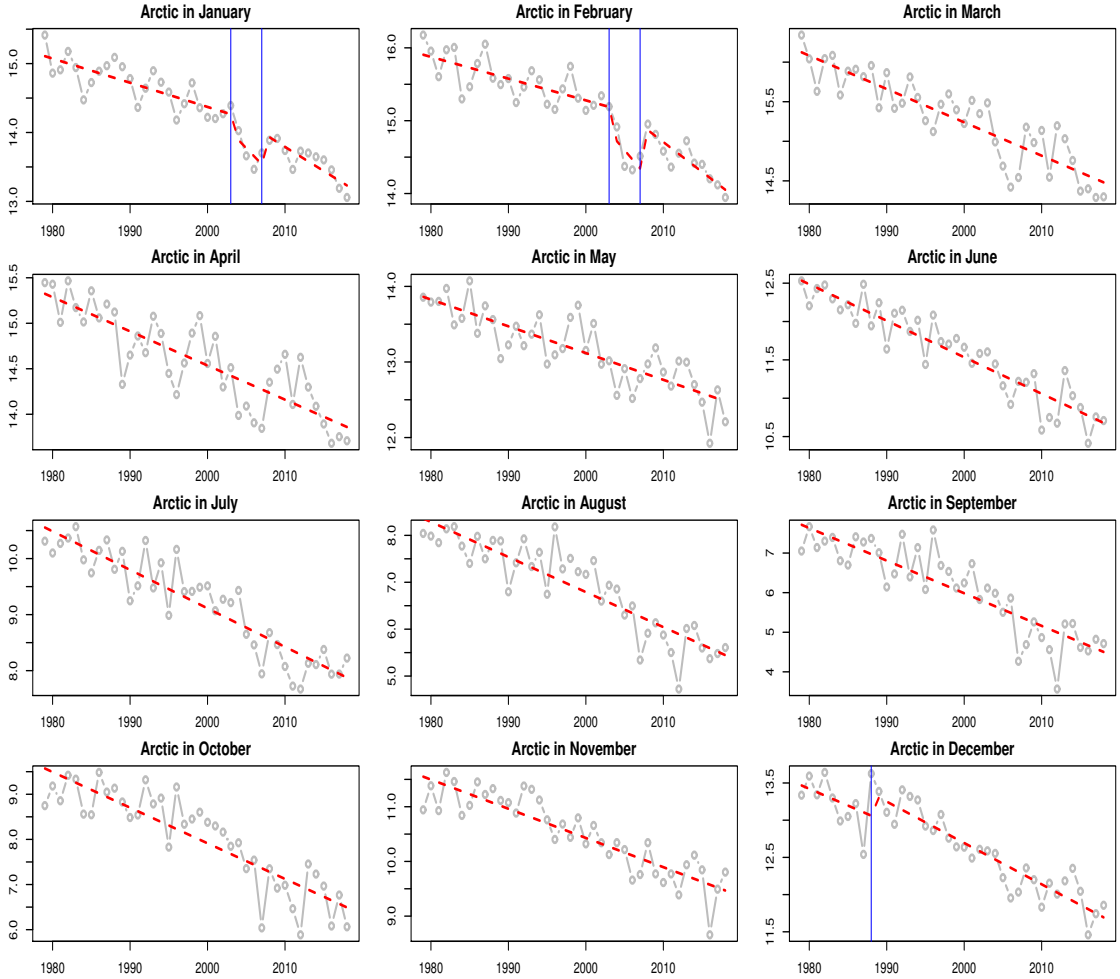


Fig. 5.10 The post-thresholded HiTS estimates of the piecewise-linear trend for monthly sea ice extent in the Arctic from 1979 to 2018 analysed in Section 5.5.2. The data series (grey dots), the HiTS estimate (----) and survived change-points (|) for each month.

5.6 Proofs

The proofs of Theorems 5.1-5.5 and Corollaries 5.1-5.4 are given below.

5.6.1 Some useful lemmas

We first present preparatory lemmas.

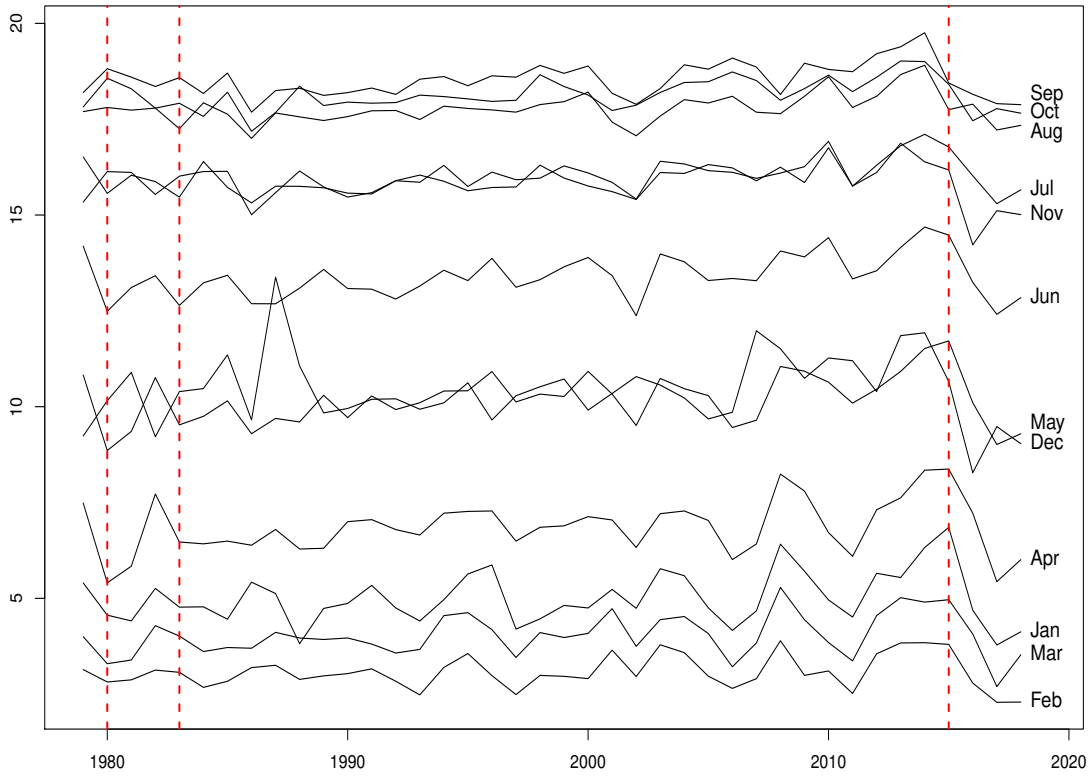


Fig. 5.11 The monthly average sea ice extent in the Antarctic from 1979 to 2018 analysed in Section 5.5.2 (——) and the estimated change-points returned by HiTS (⋮).

Lemma 5.1 *Let* $\{\boldsymbol{X}_i\}_{i=1}^n$ *follow model* (5.1) *in scenario (S1) and let Assumption 5.1 hold. We then have* $P(A_{n,T}) \geq 1 - C_3(nT)^{-1}$ *where*

$$A_{n,T} = \left\{ \max_{i,j,k} |\langle \psi^{(j,k)}, \boldsymbol{\varepsilon}_i \rangle| \leq \lambda \right\}, \tag{5.37}$$

λ *is as in Assumption 5.2 and* C_3 *is a positive constant.*

Proof. Using a simple Bonferroni inequality, we have

$$1 - P(A_{n,T}) \leq \sum_{i,j,k} P(|Z| > \lambda) \leq nT^3 \frac{\phi_Z(\lambda)}{\lambda} \leq \frac{C_3}{nT} \tag{5.38}$$

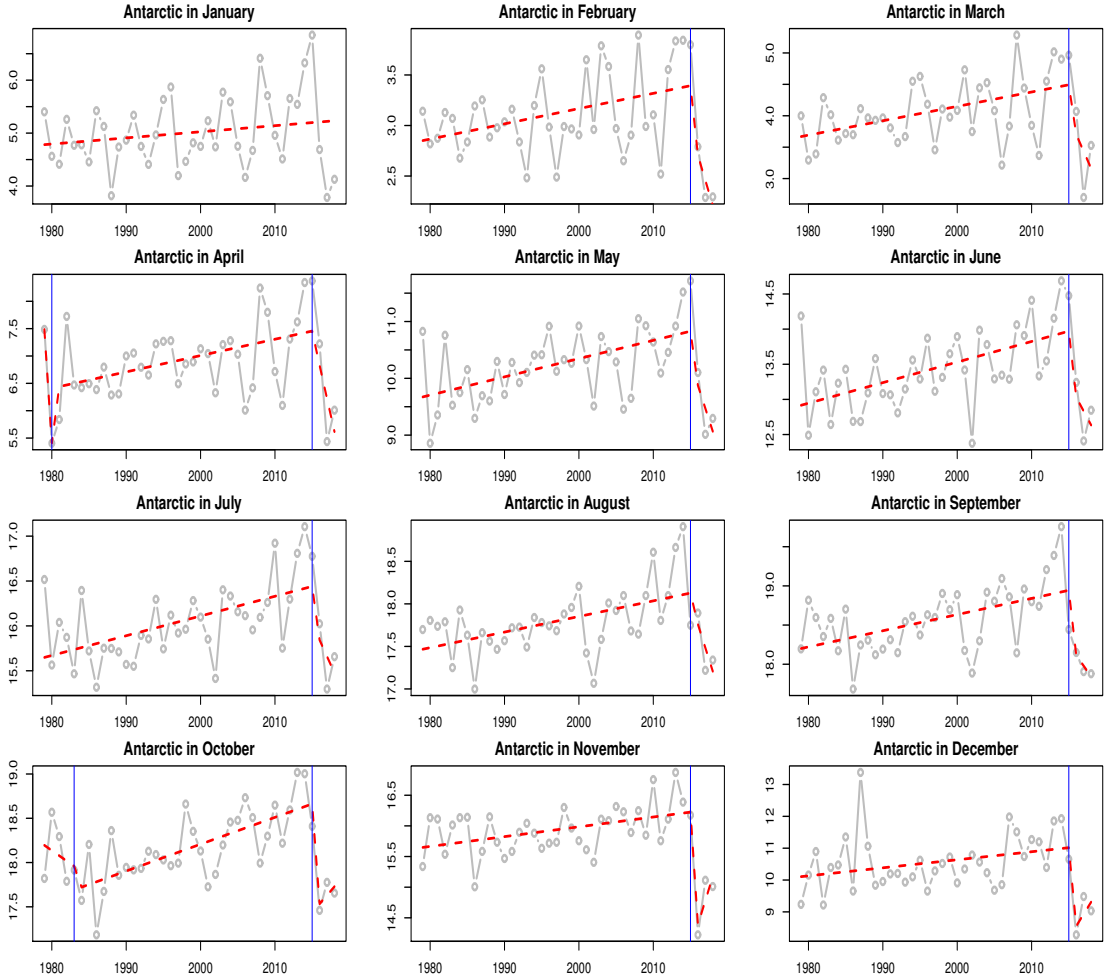


Fig. 5.12 The post-thresholded HiTS estimates of the piecewise-linear trend for monthly sea ice extent in the Antarctic from 1979 to 2018 analysed in Section 5.5.2. The data series (grey dots), the HiTS estimate (----) and survived change-points (|) for each month.

where ϕ_Z is the p.d.f. of a standard normal Z and

$$P(|Z| > \lambda) = 2\frac{1}{\sqrt{2\pi}}\int_{\lambda}^{\infty} e^{-x^2/2}dx \leq 2\frac{1}{\sqrt{2\pi}}\int_{\lambda}^{\infty}\frac{x}{\lambda}e^{-x^2/2}dx = 2\frac{e^{-\lambda^2/2}}{\lambda\sqrt{2\pi}}. \tag{5.39}$$

This completes the proof.

Lemma 5.2 *In scenario (S2), let* $\psi^{(j,k)} = \sum_{m=1}^{I^{(j,k)}} \phi_m^{(j,k)} g_m^{(j,k)}$ *where* $\phi_m^{(j,k)}$ *are constants and* $g_m^{(j,k)}$ *are vectors of equal length with* $\psi^{(j,k)}$ *where* $I^{(j,k)} \in \{3, 4\}, j = 1, \ldots, J,\ k =$

$1, \ldots, K(j)$. *If we define the set* $G = \{g_l\}$ *where there is a unique correspondence between* $\left\{g_m^{(j,k)}{}_{m=1,\ldots,I^{(j,k)},j=1,\ldots,J,\,k=1,\ldots,K(j)}\right\}$ *and* $\{g_l\}$, *we then have* $P(B_{n,T}) \geq 1 - C_3(nT)^{-1}$ *where*

$$B_{n,T} = \left\{ \max_i \max_{g_l \in G} |g_l^\top \boldsymbol{\varepsilon}_i| \leq \lambda \right\}, \tag{5.40}$$

$i = 1, \ldots, n$, λ *is as in Assumption 5.2 and* C_3 *is a positive constant.*

Proof. In Section 4.6, it is shown that there exist at most T^2 vectors g_l in the set G and for any fixed (j,k), $g_m^{(j,k)}$ and $\phi_m^{(j,k)}$ satisfy the conditions, $\left(g_m^{(j,k)}\right)^\top g_m^{(j,k)} = 1$, $\left(g_m^{(j,k)}\right)^\top g_{m'}^{(j,k)} = 0$ and $\sum_m \left(\phi_m^{(j,k)}\right)^2 = 1$. Therefore, using a simple Bonferroni inequality, we have

$$1 - P(B_{n,T}) \leq \sum_i \sum_G P(|Z| > \lambda) \leq 2nT^2 \frac{\phi_Z(\lambda)}{\lambda} \leq \frac{C_3}{nT} \tag{5.41}$$

where ϕ_Z is the p.d.f. of a standard normal Z and

$$P(|Z| > \lambda) = 2\frac{1}{\sqrt{2\pi}} \int_\lambda^\infty e^{-x^2/2} dx \leq 2\frac{1}{\sqrt{2\pi}} \int_\lambda^\infty \frac{x}{\lambda} e^{-x^2/2} dx = 2\frac{e^{-\lambda^2/2}}{\lambda\sqrt{2\pi}}. \tag{5.42}$$

This completes the proof.

Lemma 5.3 *Let* $\mathcal{S}_j^1 = \left\{1 \leq k \leq K(j) : \left\{d_{i,[p,q,r]}^{(j,k)}\right\}_{i=1}^n \text{ such that } p < \eta_\ell + 1/2 < r \text{ for some } \ell = 1, \ldots, N\right\}$, *and* $\mathcal{S}_j^0 = \{1, \ldots, K(j)\} \setminus \mathcal{S}_j^1$. *On the set* $B_{n,T}$ *which satisfies* $P(B_{n,T}) \to 1$ *as* $n, T \to \infty$, *we have*

$$\max_{\substack{i=1,\ldots,n,\\ j=1,\ldots,J,\\ k \in \mathcal{S}_j^0}} \left| d_i^{(j,k)} \right| \leq \lambda, \tag{5.43}$$

where λ *is as in Assumption 5.2 and* $B_{n,T}$ *is as in* (5.40).

Proof. On the set $B_{n,T}$, the following holds for $i = 1, \ldots, n, j = 1, \ldots, J, k \in \mathcal{S}_j^0$,

$$\begin{aligned}
\left|d_i^{(j,k)}\right| &= \left|(\psi^{(j,k)})^\top \boldsymbol{\varepsilon}_i\right| \\
&= \left|\phi_1^{(j,k)}\left(g_1^{(j,k)}\right)^\top \boldsymbol{\varepsilon}_i + \phi_2^{(j,k)}\left(g_2^{(j,k)}\right)^\top \boldsymbol{\varepsilon}_i + \phi_3^{(j,k)}\left(g_3^{(j,k)}\right)^\top \boldsymbol{\varepsilon}_i + \phi_4^{(j,k)}\left(g_4^{(j,k)}\right)^\top \boldsymbol{\varepsilon}_i\right| \\
&\leq \max_{j,\,k}\left(\left|\phi_1^{(j,k)}\right| + \left|\phi_2^{(j,k)}\right| + \left|\phi_3^{(j,k)}\right| + \left|\phi_4^{(j,k)}\right|\right) \cdot \left(\max_i \max_{l:\, g_l \in G} \left|g_l^\top \boldsymbol{\varepsilon}_i\right|\right).
\end{aligned}$$

The condition, $\sum_m \left(\phi_m^{(j,k)}\right)^2 = 1$ for any fixed (j,k), given in the proof of Lemma 5.2 implies $\max_m \left|\phi_m^{(j,k)}\right| \leq 1$ for any (j,k), thus we have (5.43) when the constant C_1 for λ in (5.43) is larger than or equal to 4 times C_1 used in (5.40).

Lemma 5.4 *Let $\{\boldsymbol{X}_i\}_{i=1}^n$ follow model (5.1) in scenario (S1) and let Assumptions 5.1 and 5.7 hold. We then have $P(C_{n,T}) \geq 1 - C_3(nT)^{-1}$ where*

$$C_{n,T} = \left\{\max_{i,j,k} |\langle \psi^{(j,k)}, \boldsymbol{\varepsilon}_i \rangle| \leq \lambda\right\}, \tag{5.44}$$

λ is as in Assumption 5.8 and C_3 is a positive constant.

Proof. Let B_i be the autocorrelation matrix of $\boldsymbol{\varepsilon}_i$ where $B_i = [\rho_i(j-k)]_{j,k=1,\ldots,T}$. Then by the argument used in the proof of Corollary 1 of Baranowski et al. (2019) (i.e. the largest eigenvalue of B_i is bounded above by $\sum_{k=-\infty}^{\infty} |\rho_i(k)|$), we attain $\|B_i\|_\infty \leq \sum_{k=-\infty}^{\infty} |\rho_i(k)|$ for $i = 1, \ldots, n$. For any fixed (j,k), $\langle \psi^{(j,k)}, \boldsymbol{\varepsilon}_i \rangle$ follows a normal distribution with mean zero and

$$\mathrm{Var}(\langle \psi^{(j,k)}, \boldsymbol{\varepsilon}_i \rangle) = (\psi^{(j,k)})^\top B_i (\psi^{(j,k)}) \leq \sum_{k=-\infty}^{\infty} |\rho_i(k)|. \tag{5.45}$$

Then we have that

$$P(\langle \psi^{(j,k)}, \boldsymbol{\varepsilon}_i \rangle \geq \lambda) = P\left(\frac{\langle \psi^{(j,k)}, \boldsymbol{\varepsilon}_i \rangle}{\sqrt{R}} \geq C_3\sqrt{2\log(nT)}\right) \leq \frac{e^{-C_3^2 \log(nT)}}{C_3\sqrt{2\log(nT)}}. \tag{5.46}$$

Considering all possible nT^3 combinations of (j,k), a simple Bonferroni inequality returns

$$1 - P(C_{n,T}) \leq \frac{C_3}{nT}. \tag{5.47}$$

This completes the proof.

Lemma 5.5 *In scenario (S2), let* $\psi^{(j,k)} = \sum_{m=1}^{I^{(j,k)}} \phi_m^{(j,k)} g_m^{(j,k)}$ *where* $\phi_m^{(j,k)}$, $g_m^{(j,k)}$ *and* $I^{(j,k)}$ *are defined as in Lemma 5.2. Let Assumptions 5.1 and 5.7 hold and the set* $G = \{g_l\}$ *is defined as in Lemma 5.2. We then have* $P(D_{n,T}) \geq 1 - C_3(nT)^{-1}$ *where*

$$D_{n,T} = \left\{ \max_i \max_{g_l \in G} |g_l^\top \boldsymbol{\varepsilon}_i| \leq \lambda \right\}, \tag{5.48}$$

λ *is as in Assumption 5.8 and* C_3 *is a positive constant.*

Proof. As in Lemma 5.2, there exist at most T^2 vectors g_l in the set G and for any fixed (j,k), $g_m^{(j,k)}$ and $\phi_m^{(j,k)}$ satisfy the conditions, $\left(g_m^{(j,k)}\right)^\top g_m^{(j,k)} = 1$, $\left(g_m^{(j,k)}\right)^\top g_{m'}^{(j,k)} = 0$ and $\sum_m \left(\phi_m^{(j,k)}\right)^2 = 1$. Let B_i be the autocorrelation matrix of $\boldsymbol{\varepsilon}_i$ where $B_i = [\rho_{j-k}]_{j,k=1,\ldots,T}$. Using the argument used in Lemma 5.4, for any fixed l, $g_l^\top \boldsymbol{\varepsilon}_i$ follows a normal distribution with mean zero and

$$\mathrm{Var}(g_l^\top \boldsymbol{\varepsilon}_i) = g_l^\top B_i g_l \leq \sum_{k=-\infty}^{\infty} |\rho_i(k)|. \tag{5.49}$$

By the same argument used in the proof of Lemma 5.4, we have

$$1 - P(D_{n,T}) \leq \frac{C_3}{nT}. \tag{5.50}$$

Lemma 5.6 *Let* $\{\boldsymbol{X}_i\}_{i=1}^n$ *follow model* (5.1) *in scenario (S1) and let Assumptions* 5.1 *and* 5.9 *hold. We then have* $P(E_{n,T}) \geq 1 - C_3(nT)^{-1}$ *where*

$$E_{n,T} = \Big\{ \max_{i,j,k} |\langle \psi^{(j,k)}, \boldsymbol{\varepsilon}_i \rangle| \leq \lambda \Big\}, \tag{5.51}$$

$\boldsymbol{\varepsilon}_i = (\varepsilon_{i,1}, \ldots, \varepsilon_{i,T})$, λ *is as in Assumption* 5.2 *and* C_3 *is a positive constant.*

Proof. For any fixed (j,k), $\left\{U_i^{(j,k)}\right\}_{i=1}^n$ forms a centred Gaussian random vector with $E\Big(\Big(U_i^{(j,k)}\Big)^2\Big) = 1$ for all $i = 1, \ldots, n$ where $U_i^{(j,k)} = \langle \psi^{(j,k)}, \boldsymbol{\varepsilon}_i \rangle$. As there exist at most T^3 basis $\psi^{(j,k)}$, we consider the set $\boldsymbol{W} = \Big\{\boldsymbol{U}^\ell = (U_1^\ell, \ldots, U_n^\ell), \ell = 1, \ldots, T^3\Big\}$ where there exist a unique correspondence between the set $\boldsymbol{W}$ and the set $\Big\{\boldsymbol{U}^{(j,k)} = (U_1^{(j,k)}, \ldots, U_n^{(j,k)}), j = 1, \ldots, J, k = 1, \ldots, K(j)\Big\}$.

We denote that the $\text{cov}(U_i^\ell, U_j^\ell)$ depends on the square of the sum of non-zero elements in the corresponding ψ and also depends on the corresponding correlation element selected from Σ. Then we can find the Gaussian random vector $\boldsymbol{U}^*$ such that $E((U_i^* - U_j^*)^2) \geq E((U_i^\ell - U_j^\ell)^2)$ in the set $\boldsymbol{W}$. Using the Slepian's inequality (Slepian, 1962), for any $a \in \mathbb{R}$ and for all ℓ,

$$P\Big(\max_i U_i^\ell > a\Big) \leq P\Big(\max_i U_i^* > a\Big). \tag{5.52}$$

Using a simple Bonferroni inequality, we have

$$1 - P(E_{n,T}) = P\Big(\max_{i,j,k} |\langle \psi^{(j,k)}, \boldsymbol{\varepsilon}_i \rangle| > \lambda\Big) \leq nT^3 P\Big(\max_i U_i^* > \lambda\Big) \leq nT^3 \frac{\phi_Z(\lambda)}{\lambda} \leq \frac{C_3}{nT}, \tag{5.53}$$

where ϕ_Z is the p.d.f. of a standard normal Z. This completes the proof.

Lemma 5.7 *Let* $\{\boldsymbol{X}_i\}_{i=1}^n$ *follow model* (5.1) *in scenario (S2). Let Assumptions* 5.1 *and* 5.9 *hold and the set* $G = \{g_l\}$ *is defined as in Lemma* 5.2. *We then have*

$P(F_{n,T}) \geq 1 - C_3(nT)^{-1}$ *where*

$$F_{n,T} = \left\{ \max_i \max_{g_l \in G} \ |g_l^\top \boldsymbol{\varepsilon}_i| \leq \lambda \right\}, \tag{5.54}$$

λ *is as in Assumption 5.2 and* C_3 *is a positive constant.*

Proof. For any fixed l, $\left\{U_i^l\right\}_{i=1}^n$ forms a centred Gaussian random vector with $E\left(\left(U_i^l\right)^2\right) = 1$ for all $i = 1, \ldots, n$ where $U_i^l = \langle g_l, \boldsymbol{\varepsilon}_i \rangle$. As shown in Lemma 5.2, there exist at most T^2 basis vector g_l in the set G. We now consider the set $\boldsymbol{V} = \left\{\boldsymbol{U}^\ell = (U_1^\ell, \ldots, U_n^\ell), \ell = 1, \ldots, T^2\right\}$. Following the same argument used in the proof of Lemma 5.6 and using the Slepian's inequality (Slepian, 1962), for any $a \in \mathbb{R}$ and for all ℓ,

$$P\left(\max_i U_i^\ell > a \right) \leq P\left(\max_i U_i^* > a \right). \tag{5.55}$$

Using a simple Bonferroni inequality, we have

$$1 - P(F_{n,T}) = P\left(\max_i \max_{g_l \in G} \ |g_l^\top \boldsymbol{\varepsilon}_i| > \lambda \right) \leq 2nT^2 P\left(\max_i U_i^* > \lambda \right) \leq 2nT^2 \frac{\phi_Z(\lambda)}{\lambda} \leq \frac{C_3}{nT}, \tag{5.56}$$

where ϕ_Z is the p.d.f. of a standard normal Z. This completes the proof.

5.6.2 Proof of Theorems 5.1 - 5.5

Proof of Theorem 5.1. Let $\mathcal{S}_j^1$ and $\mathcal{S}_j^0$ be as in Lemma 5.3. From the conditional orthonormality of the unbalanced wavelet transform, on the set $A_{n,T}$ defined in Lemma

5.1, we have

$$
\begin{aligned}
&\|\tilde{\boldsymbol{f}} - \boldsymbol{f}\|_{n,T}^2 \\
&= \frac{1}{n}\frac{1}{T}\sum_{i=1}^{n}\sum_{j=1}^{J}\sum_{k=1}^{K(j)}\left(d_i^{(j,k)}\cdot\mathbb{I}\Big\{\exists(j',k')\in\mathcal{C}_{j,k}\quad \max_i\left|d_i^{(j',k')}\right|>\lambda\Big\}-\mu_i^{(j,k)}\right)^2 \\
&\quad+\frac{1}{nT}\sum_{i=1}^{n}(s_i^{1,[1,T]}-\mu_i^{(0,1)})^2 \\
&\le \frac{1}{n}\frac{1}{T}\sum_{i=1}^{n}\sum_{j=1}^{J}\left(\sum_{k\in\mathcal{S}_j^0}+\sum_{k\in\mathcal{S}_j^1}\right)\left(d_i^{(j,k)}\cdot\mathbb{I}\Big\{\exists(j',k')\in\mathcal{C}_{j,k}\quad \max_i\left|d_i^{(j',k')}\right|>\lambda\Big\}-\mu_i^{(j,k)}\right)^2 \\
&\quad+2C_1^2T^{-1}\log(nT) \\
&=: I+II+2C_1^2T^{-1}\log(nT). \qquad (5.57)
\end{aligned}
$$

By Lemma 5.1, $\mathbb{I}\Big\{\exists(j',k')\in\mathcal{C}_{j,k}\ \max_i\left|d_i^{(j',k')}\right|>\lambda\Big\}=0$ for $k\in\mathcal{S}_j^0$ and also by the fact that $\mu_i^{(j,k)}=0$ for $i=1,\dots,n, j=1,\dots,J, k\in\mathcal{S}_j^0$, we have $I=0$. For II, we denote $\mathcal{B}=\Big\{\exists(j',k')\in\mathcal{C}_{j,k}\ \max_i\left|d_i^{(j',k')}\right|>\lambda\Big\}$ and have

$$
\begin{aligned}
&\max_i\left\{\left(d_i^{(j,k)}\cdot\mathbb{I}\Big\{\mathcal{B}\Big\}-\mu_i^{(j,k)}\right)^2\right\} \qquad (5.58)\\
&=\max_i\left\{\left(d_i^{(j,k)}\cdot\mathbb{I}\Big\{\mathcal{B}\Big\}-d_i^{(j,k)}+d_i^{(j,k)}-\mu_i^{(j,k)}\right)^2\right\} \\
&\le\max_i\left\{\left(d_i^{(j,k)}\right)^2\cdot\mathbb{I}\Big(\max_i\left|d_i^{(j',k')}\right|\le\lambda\Big)+\left(d_i^{(j,k)}-\mu_i^{(j,k)}\right)^2\right. \\
&\quad\left.+2\left|d_i^{(j,k)}\right|\cdot\mathbb{I}\Big(\max_i\left|d_i^{(j',k')}\right|\le\lambda\Big)\cdot\left|d_i^{(j,k)}-\mu_i^{(j,k)}\right|\right\} \\
&\le\lambda^2+2C_1^2\log(nT)+2\lambda C_1\{2\log(nT)\}^{1/2} \\
&\le 8C_1^2\log(nT).
\end{aligned}
$$

Combining with the upper bound of J, $\lceil \log(T)/\log(1-\rho)^{-1} \rceil$, and the fact that $|\mathcal{S}_j^1| \leq N$, we have $II \leq 8C_1^2 N T^{-1} \lceil \log(T)/\log(1-\rho)^{-1} \rceil \log(nT)$, and therefore

$$\|\tilde{\boldsymbol{f}} - \boldsymbol{f}\|_{n,T}^2 \;\leq\; C_1^2 \, \frac{1}{T} \, \log(nT) \left\{ 2 + 8N \, \lceil \, \log(T)/\log(1-\rho)^{-1} \, \rceil \right\}. \tag{5.59}$$

Also, at each scale, the estimated change-points are obtained up to size N, combining it with the largest scale J, the number of change-points in $\tilde{\boldsymbol{f}}$ returned from the inverse HiTGUW transformation is up to $CN\log(T)$ where C is a constant.

Proof of Theorem 5.2. Let $\mathcal{S}_j^1$ and $\mathcal{S}_j^0$ be as in Lemma 5.3. From the conditional orthonormality of the unbalanced wavelet transform, on the set $B_{n,T}$ defined in Lemma 5.2, we have

$$\begin{aligned}
&\|\tilde{\boldsymbol{f}} - \boldsymbol{f}\|_{n,T}^2 \\
&= \frac{1}{n}\frac{1}{T} \sum_{i=1}^{n} \sum_{j=1}^{J} \sum_{k=1}^{K(j)} \left(d_i^{(j,k)} \cdot \mathbb{I}\Big\{ \exists (j',k') \in \mathcal{C}_{j,k} \quad \max_i \left| d_i^{(j',k')} \right| > \lambda \Big\} - \mu_i^{(j,k)} \right)^2 \\
&\quad + \frac{1}{nT} \sum_{i=1}^{n} (s_i^{1,[1,T]} - \mu_i^{(0,1)})^2 + \frac{1}{nT} \sum_{i=1}^{n} (s_i^{2,[1,T]} - \mu_i^{(0,2)})^2 \\
&\leq \frac{1}{n}\frac{1}{T} \sum_{i=1}^{n} \sum_{j=1}^{J} \left(\sum_{k \in \mathcal{S}_j^0} + \sum_{k \in \mathcal{S}_j^1} \right) \left(d_i^{(j,k)} \cdot \mathbb{I}\Big\{ \exists (j',k') \in \mathcal{C}_{j,k} \quad \max_i \left| d_i^{(j',k')} \right| > \lambda \Big\} - \mu_i^{(j,k)} \right)^2 \\
&\quad + 4C_1^2 T^{-1} \log(nT) \\
&=: I + II + 4C_1^2 T^{-1} \log(nT).
\end{aligned} \tag{5.60}$$

By Lemma 5.3, $\mathbb{I}\Big\{ \exists (j',k') \in \mathcal{C}_{j,k} \;\; \max_i \left| d_i^{(j',k')} \right| > \lambda \Big\} = 0$ for $k \in \mathcal{S}_j^0$ and also by the fact that $\mu_i^{(j,k)} = 0$ for $i = 1, \ldots, n, j = 1, \ldots, J, k \in \mathcal{S}_j^0$, we have $I = 0$. Through the same reasoning used in the proof of Theorem 5.2, we have

$$\max_i \left\{ \left(d_i^{(j,k)} \cdot \mathbb{I}\Big\{ \mathcal{B} \Big\} - \mu_i^{(j,k)} \right)^2 \right\} \leq 8C_1^2 \log(nT), \tag{5.61}$$

for II and

$$\|\tilde{\boldsymbol{f}} - \boldsymbol{f}\|_{n,T}^2 \leq C_1^2 \frac{1}{T} \log(nT) \left\{ 4 + 8N \left\lceil \log(T)/\log\left((1-\rho)^{-1}\right) + \log(2)/\log(1-\rho) \right\rceil \right\}. \tag{5.62}$$

Also, the number of change-points in $\tilde{\boldsymbol{f}}$ returned from the inverse HiTGUW transformation is up to $CN\log(T)$ which is equal to the one in the proof of Theorem 4.1 where C is a constant. This completes the proof.

Proof of Theorem 5.3. Let $\tilde{B}$ and $\tilde{\tilde{B}}$ the unbalanced wavelet basis corresponding to $\tilde{\boldsymbol{f}}$ and $\tilde{\tilde{\boldsymbol{f}}}$, respectively. As the change-points in $\tilde{\tilde{\boldsymbol{f}}}$ are a subset of those in $\tilde{\boldsymbol{f}}$, establishing $\tilde{\tilde{\boldsymbol{f}}}$ can be regarded as applying the HiTGUW transform again to $\tilde{\boldsymbol{f}}$, which is just a repetition of the estimation procedure $\tilde{\boldsymbol{f}}$ but performed in a greedy way. Thus $\tilde{\tilde{B}}$ is classified into two categories, 1) all basis vectors $\psi^{(j,k)} \in \tilde{B}$ such that $\psi^{(j,k)}$ is not associated with the change-points in $\tilde{\boldsymbol{f}}$ and $\max_i \left|\langle \boldsymbol{X}_i, \psi^{(j,k)} \rangle\right| = \max_i \left|d_i^{(j,k)}\right| < \lambda$ and 2) all vectors $\psi^{(j,1)}$ produced in Stage 1 of post-processing.

We now investigate how many scales are used for this particular transform. Firstly, the detail coefficients $\{d_i^{(j,k)}\}_{i=1}^n$ corresponding to the basis vectors $\psi^{(j,k)} \in \tilde{B}$ live on no more than $J = O(\log(T))$ scales and we have $|\mathcal{S}_j^1| \leq N$ by the argument used in the proofs of Theorems 5.1 and 5.2. In addition, the vectors $\psi^{(j,1)}$ in the second category above correspond to different change-points in $\tilde{\boldsymbol{f}}$ and there exist at most $\tilde{N} = O(N\log(T))$ change-points in $\tilde{\boldsymbol{f}}$ which we examine one at once (i.e. $|\mathcal{S}_j^1| \leq 1$), thus at most $\tilde{N}$ scales are required for $\{d_i^{(j,1)}\}_{i=1}^n$. Combining the results of the two categories, the equivalent of quantity II in the proofs of Theorems 5.1 and 5.2 for $\tilde{\tilde{\boldsymbol{f}}}$ is bounded by $II \leq CNT^{-1}\log(T)\log(nT)$ and this completes the proof of the l_2 result, $\left\|\tilde{\tilde{\boldsymbol{f}}} - \boldsymbol{f}\right\|_T^2 = O\left(NT^{-1}\log(T)\log(nT)\right)$ where C is a large enough positive constant.

Finally, we show that there exist at most two change-points in $\tilde{\tilde{\boldsymbol{f}}}$ between true change-points $(\eta_\ell, \eta_{\ell+1})$ for $\ell = 0, \ldots, N$ where $\eta_0 = 0$ and $\eta_{N+1} = T$. Consider

the case where three change-point for instance $(\tilde{\tilde{\eta}}_l, \tilde{\tilde{\eta}}_{l+1}, \tilde{\tilde{\eta}}_{l+2})$ lie between a pair of true change-points, (η_i, η_{i+1}). In this case, by Lemmas 5.1 and 5.3, the maximum magnitude of two detail coefficients computed from the adjacent intervals, $[\tilde{\tilde{\eta}}_l + 1, \tilde{\tilde{\eta}}_{l+1}]$ and $[\tilde{\tilde{\eta}}_{l+1} + 1, \tilde{\tilde{\eta}}_{l+2}]$, is less than λ and $\tilde{\tilde{\eta}}_{l+1}$ would get removed from the set of estimated change-points. This leads to $\tilde{\tilde{N}} \leq 2(N+1)$.

Proof of Theorem 5.4. From the assumptions of Theorem 5.4, the followings hold.

- Given any $\epsilon > 0$ and $C > 0$, for some T_1, n_1 and all $T > T_1$ and $n > n_1$, it holds that $\mathbb{P}\Big(\big\| \tilde{\tilde{\boldsymbol{f}}} - \boldsymbol{f} \big\|_{n,T}^2 > \frac{C}{4} R_{n,T} \Big) \leq \epsilon$ where $\tilde{\tilde{\boldsymbol{f}}}$ is the estimated signal in (5.29).
- For some T_2, n_2 and all $T > T_2$ and $n > n_2$, it holds that $CnTR_{n,T}(\underline{f}_{n,T}^{\ell})^{-2} < \delta_{n,T}^{\ell}$ for all $\ell = 1, \ldots, N$.

Similar to the argument used in the proof of Theorem 8 in Lin et al. (2016), we take $T \geq \max\{T_1, T_2\}$ and $n \geq \max\{n_1, n_2\}$, and let $r_{n,T}^{\ell} = \lfloor CnTR_{n,T}(\underline{f}_{n,T}^{\ell})^{-2} \rfloor$ for $\ell = 1, \ldots, N$. Suppose that there exists at least one η_ℓ whose closest estimated change-point is not within the distance of $r_{n,T}^{\ell}$. Then there is no estimated change-point in $\tilde{\tilde{\boldsymbol{f}}}$ within $r_{n,T}^{\ell}$ of η_ℓ which means that $\tilde{\tilde{f}}_{i,j}$ displays a constant function over the entire segment $j \in \{\eta_\ell - r_{n,T}^{\ell}, \ldots, \eta_\ell + r_{n,T}^{\ell}\}$ for all $i = 1, \ldots, n$. Hence

$$\frac{1}{nT} \sum_{i \in \Omega_\ell} \sum_{j=\eta_\ell - r_{n,T}^{\ell}}^{\eta_\ell + r_{n,T}^{\ell}} (\tilde{\tilde{f}}_{i,j} - f_{i,j})^2 \geq \frac{r_{n,T}^{\ell}}{2nT} \big(\underline{f}_{n,T}^{\ell} \big)^2 > \frac{C}{4} R_{n,T}. \tag{5.63}$$

We see that assuming that at least one η_ℓ does not have an estimated change-point within the distance of $r_{n,T}^{\ell}$ implies the estimation error exceeds $\frac{C}{4} R_{n,T}$ which is a contradiction as it is an event that we know occurs with probability at most ϵ. Therefore, there must exist at least one estimated change-point within the distance of $r_{n,T}^{\ell}$ from each true change-point η_ℓ where $\ell = 1, \ldots, N$.

Throughout Stage 2 of post-processing, $\tilde{\tilde{\eta}}_{\ell_0}$ is either the closest estimated change-point of any η_ℓ or not. If $\tilde{\tilde{\eta}}_{\ell_0}$ is not the closest estimated change-point to the nearest true change-point on either its left or its right, by the construction of detail coefficient in Stage 2 of post-processing, Lemma 5.1 guarantees that the corresponding detail coefficient has the magnitude less than λ and $\tilde{\tilde{\eta}}_{\ell_0}$ gets removed. Suppose $\tilde{\tilde{\eta}}_{\ell_0}$ is the closest estimated change-point of a true change-point η_ℓ and it is within the distance of $CnTR_{n,T}\left(\underline{f}^{\ell}_{n,T}\right)^{-2}$ from η_ℓ. If the corresponding detail coefficient has the magnitude less than λ and $\tilde{\tilde{\eta}}_{\ell_0}$ is removed, there must exist another $\tilde{\tilde{\eta}}_{\ell}$ within the distance of $CnTR_{n,T}\left(\underline{f}^{\ell}_{n,T}\right)^{-2}$ from η_ℓ. If there are no such $\tilde{\tilde{\eta}}_{\ell}$, then by the construction of the detail coefficient, the order of magnitude of $\max_i \left|d_i^{[p_{\ell_0}, q_{\ell_0}, r_{\ell_0}]}\right|$ would be such that $\max_i \left|d_i^{[p_{\ell_0}, q_{\ell_0}, r_{\ell_0}]}\right| > \lambda$ thus $\tilde{\tilde{\eta}}_{\ell_0}$ would not get removed. Therefore, after Stage 2 post-processing is finished, each true change-point η_ℓ has its unique estimator within the distance of $CnTR_{n,T}\left(\underline{f}^{\ell}_{n,T}\right)^{-2}$.

Proof of Theorem 5.5. From the assumptions of Theorem 5.5, the followings hold.

- Given any $\epsilon > 0$ and $C > 0$, for some T_1, n_1 and all $T > T_1$ and $n > n_1$, it holds that $\mathbb{P}\left(\left\|\tilde{\tilde{\boldsymbol{f}}} - \boldsymbol{f}\right\|_{n,T}^2 > \frac{C^3}{4} R_{n,T}\right) \leq \epsilon$ where $\tilde{\tilde{\boldsymbol{f}}}$ is the estimated signal in (5.30).
- For some T_2, n_2 and all $T > T_2$ and $n > n_2$, it holds that $C^{1/3} n^{1/3} T^{1/3} R_{n,T}^{1/3} (\underline{f}^{\ell}_{n,T})^{-2/3} < \delta^{\ell}_{n,T}$ for all $\ell = 1, \ldots, N$.

Similar to the argument used in the proof of Theorem 19 in Lin et al. (2016), we take $T \geq \max\{T_1, T_2\}$ and $n \geq \max\{n_1, n_2\}$, and let $r^{\ell}_{n,T} = \lfloor C^{1/3} n^{1/3} T^{1/3} R_{n,T}^{1/3} (\underline{f}^{\ell}_{n,T})^{-2/3} \rfloor$ for $\ell = 1, \ldots, N$. Suppose that there exist at least one η_ℓ whose closest estimated change-point is not within the distance of $r^{\ell}_{n,T}$. Then there is no estimated change-point in $\tilde{\tilde{\boldsymbol{f}}}$ within $r^{\ell}_{n,T}$ of η_ℓ which means that $\tilde{\tilde{f}}_{i,j}$ displays a linear trend over the entire

segment $j \in \{\eta_\ell - r^\ell_{n,T}, \ldots, \eta_\ell + r^\ell_{n,T}\}$ for all $i = 1, \ldots, n$. Hence

$$\frac{1}{nT} \sum_{i \in \Omega_\ell} \sum_{j=\eta_\ell - r^\ell_{n,T}}^{\eta_\ell + r^\ell_{n,T}} (\tilde{\tilde{f}}_{i,j} - f_{i,j})^2 \geq \frac{13(r^\ell_{n,T})^3}{24nT} \left(\underline{f}^\ell_{n,T}\right)^2 > \frac{C^3}{4} R_{n,T}. \tag{5.64}$$

The first inequality holds due to the Lemma 20 in Lin et al. (2016), and the second holds by the definition of $r^\ell_{n,T}$. Following the similar argument used in the proof of Theorem 5.4, there must exist at least one estimated change-point within the distance of $r^\ell_{n,T}$ from each true change-point η_ℓ and after the Stage 2 post-processing is finished, each true change-point η_ℓ has its unique estimator within the distance of $Cn^{1/3}T^{1/3}R^{1/3}_{n,T}\left(\underline{f}^\ell_{n,T}\right)^{-2/3}$.

5.6.3 Proof of Corollaries 5.1 - 5.4

Proof of Corollary 5.1. Let $\mathcal{S}^1_j$ and $\mathcal{S}^0_j$ be as in Lemma 5.3. Following the same argument used in the proof of Theorem 5.1 with Lemma 5.4, in terms of the l_2 consistency, we have

$$\|\tilde{\boldsymbol{f}} - \boldsymbol{f}\|^2_{n,T} \;\leq\; C_3^2 \, \frac{1}{T} \, R \log(nT) \left\{ 2 + 8N \, \lceil \, \log(T) / \log(1-\rho)^{-1} \, \rceil \right\}, \tag{5.65}$$

with probability approaching to 1 as $n, T \to \infty$ and the piecewise-constant estimator $\left\{\tilde{\boldsymbol{f}}_i\right\}_{i=1}^n$ in (5.27) contains $\tilde{N} \leq CN \log(T)$ change-points where C is a constant. This completes the proof of the first part corresponding to Theorem 5.1. And the conclusions of Theorems 5.3 and 5.4 are obtained by using the arguments used in the proofs of Theorems 5.3 and 5.4.

Proof of Corollary 5.2. Let $\mathcal{S}^1_j$ and $\mathcal{S}^0_j$ be as in Lemma 5.3. Following the argument used in Lemma 5.3, on the set $D_{n,T}$ which satisfies $P(D_{n,T}) \to 1$ as $n, T \to \infty$, we have $\max_{i=1,\ldots,n, j=1,\ldots,J, k \in \mathcal{S}^0_j} \left| d_i^{(j,k)} \right| \leq \lambda$ where $D_{n,T}$ is defined in Lemma 5.5. Then following

the same argument used in the proof of Theorem 5.2 with Lemma 5.5, in terms of the l_2 consistency, we have

$$\|\tilde{\boldsymbol{f}}-\boldsymbol{f}\|_{n,T}^2 \;\leq\; C_3^2\,\frac{1}{T}\,R\log(nT)\left\{4+8N\left\lceil\log(T)/\log\left((1-\rho)^{-1}\right)+\log(2)/\log(1-\rho)\right\rceil\right\}, \tag{5.66}$$

with probability approaching to 1 as $n, T \to \infty$ and the piecewise-linear estimator $\left\{\tilde{\boldsymbol{f}}_i\right\}_{i=1}^n$ in (5.28) contains $\tilde{N} \leq CN\log(T)$ change-points where C is a constant. This completes the proof of the first part corresponding to Theorem 5.2 and the conclusions of Theorems 5.3 and 5.5 are achieved by following the arguments used in the proofs of Theorems 5.3 and 5.5.

Proof of Corollary 5.3. Let $\mathcal{S}_j^1$ and $\mathcal{S}_j^0$ be as in Lemma 5.3. We attain the l_2 results by following the arguments used in the proof of Theorem 5.1 with Lemma 5.6. And the conclusions of Theorems 5.3 and 5.4 are obtained by using the arguments used in the proofs of Theorems 5.3 and 5.4.

Proof of Corollary 5.4. Let $\mathcal{S}_j^1$ and $\mathcal{S}_j^0$ be as in Lemma 5.3. Following the argument used in Lemma 5.3, on the set $F_{n,T}$ which satisfies $P(F_{n,T}) \to 1$ as $n, T \to \infty$, we have $\max_{i=1,\ldots,n,j=1,\ldots,J,k\in\mathcal{S}_j^0}\left|d_i^{(j,k)}\right| \leq \lambda$ where $F_{n,T}$ is defined in Lemma 5.7. Then, the l_2 result is attained by using the arguments used in the proof of Theorem 5.2 with Lemma 5.7. And the conclusions of Theorems 5.3 and 5.5 are achieved by following the arguments used in the proofs of Theorems 5.3 and 5.5.

Chapter 6

Conclusions

This thesis considers adaptive multiscale approaches to the trend segmentation of data sequences and linear regression. In this chapter, we provide a brief summary of our main contributions in Chapters 3, 4 and 5 and discuss possible directions for future research.

Chapter 3 introduces the smooth-rough partition model, a new way of regularising linear regression coefficients for modelling temporal dependence in random functions. The SRP model represents a compromise between a completely unregularised and a completely regularised linear model in that it keeps all the effects as non-zero but partitions them into two classes of regularity. The SRP framework can be generalised to linear regression with a scalar response Y and a discretised functional predictor $X(t)$, and here are some interesting avenues to apply. The SRP approach can be a useful alternative to sparsity-based approaches as retaining the smooth non-zero regression parameter can be beneficial for prediction, as demonstrated in Chapter 3. Especially, when potential regressors have been pre-ordered in terms of their importance, the SRP framework can replace truncation or cutting-off techniques. For example, when a principal component (PC) regression is carried out, the SRP idea allows us to keep the entire PC scores under two different smoothness constraints rather than removing

most of them by truncation, by estimating the change-point of the effect of PC scores in terms of extent and smoothness. In a general linear regression, the SRP framework would be a useful tool if one pursues the balance of prediction and interpretability, as it keeps all the regression parameters for a better prediction performance but also gives a reasonable interpretation by estimating a change-point.

In Chapter 4, we propose TrendSegment, a methodology for detecting multiple change-points corresponding to linear trend changes or point anomalies in univariate time series. We first consider the situation when the underlying signal of the data becomes more complex, e.g. a mixture of constant, linear and quadratic trends, in which case our method TrendSegment can be extended to offer a multi-trend segmentation. In the simulations performed in Chapter 4, TrendSegment performs pretty well in a mix of piecewise-constant and piecewise-linear signals, however it gives a piecewise-linear estimate instead of distinguishing the intervals of the linear trend from the constant ones. To examine which polynomial trend is appropriate for each subregion of the TrendSegment estimate, we can think of simultaneous investigation of the filter for each of constancy, linearity and quadraticity and the corresponding detail coefficients, where the filters are operated along with the tree structure constructed to fit a piecewise function with the highest order of interest.

Another possible extension of the TrendSegment procedure is to propose a hybrid method of top-down and bottom-up transforms for trend segmentation. The simulation studies in Chapter 4 show that the bottom-up approach performs well in estimating the number of change-point but is less attractive than competitors in localisation (i.e. estimating the exact locations of change-points). This is due to the fact that the bottom-up transform is constructed in a way of focusing on local features in its early stages and on global features next. The hybrid approach will promote detection of

change-points across a wider range of signals and will improve the estimation accuracy of change-point by taking advantages of both top-down and bottom-up transforms.

Chapter 5 introduces High-dimensional Trend Segmentation (HiTS), a methodology for detecting trend changes in high-dimensional panel data, which extends the bottom-up transformation proposed in Chapter 4 into high-dimensional settings. The HiTS procedure can be extended in several directions. The analysis of South Africa temperature data in Section 5.5.1 implies that the feature extraction results of the HiTS procedure can pave the way for time series clustering. A similar way of thinking appears in previous works, e.g. Jirak (2015) studies a way of identifying the set of coordinates those undergo a change, however an extension to a higher-level representation such as classification or clustering has not previously been studied. Another avenue to extend this work is relaxing those assumptions on temporal and cross-sectional dependences stated in Section 5.3, to make the HiTS procedure work in more general noise settings.

References

Akaike, H. (1974). A new look at the statistical model identification. In *Selected Papers of Hirotugu Akaike*, pages 215–222. Springer. 64

Anastasiou, A. and Fryzlewicz, P. (2019). Detecting multiple generalized change-points by isolating single ones. *arXiv preprint arXiv:1901.10852*. 33, 41, 128

Aneiros-Pérez, G. and Vieu, P. (2008). Nonparametric time series prediction: A semi-functional partial linear modeling. *Journal of Multivariate Analysis*, 99:834–857. 28

Antoniadis, A., Paparoditis, E., and Sapatinas, T. (2006). A functional wavelet–kernel approach for time series prediction. *Journal of the Royal Statistical Society Series B*, 68:837–857. 21

Asgharian, M. and Wolfson, D. B. (2001). Covariates in multipath change-point problems: Modelling and consistency of the mle. *Canadian Journal of Statistics*, 29:515–528. 47

Aston, J. A. and Kirch, C. (2018). High dimensional efficiency with applications to change point tests. *Electronic Journal of Statistics*, 12:1901–1947. 53

Aue, A., Hörmann, S., Horváth, L., and Reimherr, M. (2009). Break detection in the covariance structure of multivariate time series models. *The Annals of Statistics*, 37:4046–4087. 49

Aue, A., Norinho, D. D., and Hörmann, S. (2015). On the prediction of stationary functional time series. *Journal of the American Statistical Association*, 110:378–392. 21

Bai, J. (2010). Common breaks in means and variances for panel data. *Journal of Econometrics*, 157:78–92. 50

Bai, J. and Perron, P. (1998). Estimating and testing linear models with multiple structural changes. *Econometrica*, 66:47–78. 39

Bai, J. and Perron, P. (2003). Computation and analysis of multiple structural change models. *Journal of Applied Econometrics*, 18:1–22. 40, 129

Bandi, F. M. and Phillips, P. C. (2003). Fully nonparametric estimation of scalar diffusion models. *Econometrica*, 71:241–283. 85

Baranowski, R., Chen, Y., and Fryzlewicz, P. (2019). Narrowest-over-threshold detection of multiple change points and change-point-like features. *Journal of the Royal Statistical Society Series B*, 81:649–672. 32, 33, 41, 128, 219

Bardwell, L. and Fearnhead, P. (2017). Bayesian detection of abnormal segments in multiple time series. *Bayesian Analysis*, 12:193–218. 29, 38

Bardwell, L., Fearnhead, P., Eckley, I. A., Smith, S., and Spott, M. (2019). Most recent changepoint detection in panel data. *Technometrics*, 61:88–98. 43, 49

Barigozzi, M., Cho, H., and Fryzlewicz, P. (2018). Simultaneous multiple change-point and factor analysis for high-dimensional time series. *Journal of Econometrics*, 206:187–225. 53

Barry, D. and Hartigan, J. A. (1993). A bayesian analysis for change point problems. *Journal of the American Statistical Association*, 88:309–319. 38

Bélisle, P., Joseph, L., MacGibbon, B., Wolfson, D. B., and Du Berger, R. (1998). Change-point analysis of neuron spike train data. *Biometrics*, 54:113–123. 46

Bhattacharjee, M., Banerjee, M., and Michailidis, G. (2019). Change point estimation in panel data with temporal and cross-sectional dependence. *arXiv preprint arXiv:1904.11101*. 50, 52

Booth, N. and Smith, A. (1982). A bayesian approach to retrospective identification of change-points. *Journal of Econometrics*, 19:7–22. 44

Bosq, D. (2000). *Linear Processes in Function Spaces*. New York: Springer-Verlag. 21

Boysen, L., Kempe, A., Liebscher, V., Munk, A., and Wittich, O. (2009). Consistencies and rates of convergence of jump-penalized least squares estimators. *The Annals of Statistics*, 37:157–183. 30

Broemeling, L. (1972). Bayesian procedures for detecting a change in a sequence of random variables. *Metron*, 30:1–14. 38

Cai, T. T. and Hall, P. (2006). Prediction in functional linear regression. *The Annals of Statistics*, 34:2159–2179. 71

Candes, E. and Tao, T. (2007). The dantzig selector: Statistical estimation when p is much larger than n. *The Annals of Statistics*, 35:2313–2351. 24

Cardot, H., Crambes, C., Kneip, A., and Sarda, P. (2007). Smoothing splines estimators in functional linear regression with errors-in-variables. *Computational Statistics and Data Analysis*, 51:4832–4848. 62, 70

Cardot, H., Ferraty, F., and Sarda, P. (2003). Spline estimators for the functional linear model. *Statistica Sinica*, 13:571–591. 62, 70, 73

Chandola, V., Banerjee, A., and Kumar, V. (2009). Anomaly detection: A survey. *ACM computing surveys (CSUR)*, 41:Article 15. 37

Chen, H. and Zhang, N. (2015). Graph-based change-point detection. *The Annals of Statistics*, 43:139–176. 53

Chen, K., Delicado, P., and Müller, H.-G. (2017). Modelling function-valued stochastic processes, with applications to fertility dynamics. *Journal of the Royal Statistical Society Series B*, 79:177–196. 21

Chernoff, H. and Zacks, S. (1964). Estimating the current mean of a normal distribution which is subjected to changes in time. *The Annals of Mathematical Statistics*, 35:999–1018. 38

Cho, H. (2016). Change-point detection in panel data via double cusum statistic. *Electronic Journal of Statistics*, 10:2000–2038. 51, 52, 192

Cho, H. and Fryzlewicz, P. (2012). Multiscale and multilevel technique for consistent segmentation of nonstationary time series. *Statistica Sinica*, 22:207–229. 34

Cho, H. and Fryzlewicz, P. (2015). Multiple-change-point detection for high dimensional time series via sparsified binary segmentation. *Journal of the Royal Statistical Society Series B*, 77:475–507. 51, 52, 171, 192

Comiso, J. C., Gersten, R. A., Stock, L. V., Turner, J., Perez, G. J., and Cho, K. (2017). Positive trend in the antarctic sea ice cover and associated changes in surface temperature. *Journal of Climate*, 30:2251–2267. 142

Crambes, C., Kneip, A., and Sarda, P. (2009). Smoothing splines estimators for functional linear regression. *The Annals of Statistics*, 37:35–72. 62, 70, 71

Cribben, I. and Yu, Y. (2017). Estimating whole-brain dynamics by using spectral clustering. *Journal of the Royal Statistical Society Series C*, 66:607–627. 16, 43, 53

Crosier, R. B. (1988). Multivariate generalizations of cumulative sum quality-control schemes. *Technometrics*, 30:291–303. 45

Dalalyan, A. S., Hebiri, M., and Lederer, J. (2017). On the prediction performance of the lasso. *Bernoulli*, 23:552–581. 126

Descary, M.-H. and Panaretos, V. M. (2019). Functional data analysis by matrix completion. *The Annals of Statistics*, 47:1–38. 70

Dette, H. and Gösmann, J. (2018). Relevant change points in high dimensional time series. *Electronic Journal of Statistics*, 12:2578–2636. 43

Eichinger, B. and Kirch, C. (2018). A mosum procedure for the estimation of multiple random change points. *Bernoulli*, 24:526–564. 37

Enikeeva, F. and Harchaoui, Z. (2013). High-dimensional change-point detection with sparse alternatives. *arXiv preprint arXiv:1312.1900*. 51, 52

Ertel, J. E. and Fowlkes, E. B. (1976). Some algorithms for linear spline and piecewise multiple linear regression. *Journal of the American Statistical Association*, 71:640–648. 42

Fan, J. and Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, 96:1348–1360. 25

Fearnhead, P. (2006). Exact and efficient bayesian inference for multiple changepoint problems. *Statistics and computing*, 16:203–213. 38

Fearnhead, P. and Liu, Z. (2007). On-line inference for multiple changepoint problems. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69:589–605. 38

Fearnhead, P. and Rigaill, G. (2019). Changepoint detection in the presence of outliers. *Journal of the American Statistical Association*, 114:169–183. 37

Ferraty, F., Hall, P., and Vieu, P. (2010). Most-predictive design points for functional data predictors. *Biometrika*, 97:807–824. 21, 26, 74

Ferraty, F. and Vieu, P. (2002). The functional nonparametric model and application to spectrometric data. *Computational Statistics*, 17:545–564. 74

Fisch, A. T. M., Eckley, I. A., and Fearnhead, P. (2018). A linear time method for the detection of point and collective anomalies. *arXiv preprint arXiv:1806.01947*. 16, 29, 37, 143

Frick, K., Munk, A., and Sieling, H. (2014). Multiscale change point inference. *Journal of the Royal Statistical Society Series B*, 76:495–580. 30

Fryzlewicz, P. (2007). Unbalanced haar technique for nonparametric function estimation. *Journal of the American Statistical Association*, 102:1318–1327. 33

Fryzlewicz, P. (2014). Wild binary segmentation for multiple change-point detection. *The Annals of Statistics*, 42:2243–2281. 32, 51, 157

Fryzlewicz, P. (2018a). Detecting possibly frequent change-points: Wild binary segmentation 2 and steepest-drop model selection. *arXiv preprint arXiv:1812.06880*. 32

Fryzlewicz, P. (2018b). Tail-greedy bottom-up data decompositions and fast mulitple change-point detection. *The Annals of Statistics*, 46:3390–3421. 34, 37, 98, 106, 114, 126, 157, 158, 171, 179

Fryzlewicz, P. and Subba Rao, S. (2014). Multiple-change-point detection for auto-regressive conditional heteroscedastic processes. *Journal of the Royal Statistical Society series B*, 76:903–924. 34

Gabrys, R., Horváth, L., and Kokoszka, P. (2010). Tests for error correlation in the functional linear model. *Journal of the American Statistical Association*, 105:1113–1125. 21

Goia, A. (2012). A functional linear model for time series prediction with exogenous variables. *Statistics and Probability Letters*, 82:1005–1011. 28

Goia, A. and Vieu, P. (2015). A partitioned single functional index model. *Computational Statistics*, 30:673–692. 27

Groen, J. J., Kapetanios, G., and Price, S. (2013). Multivariate methods for monitoring structural change. *Journal of Applied Econometrics*, 28:250–274. 16, 43, 49

Hall, P. and Hooker, G. (2016). Truncated linear models for functional data. *Journal of the Royal Statistical Society Series B*, 78:637–653. 26, 27, 70, 71, 90, 94

Hampel, F. R. (1974). The influence curve and its role in robust estimation. *Journal of the American Statistical Association*, 69:383–393. 124, 127, 183, 188

Han, K., Müller, H.-G., and Park, B. U. (2018). Smooth backfitting for additive modeling with small errors-in-variables, with an application to additive functional regression for multiple predictor functions. *Bernoulli*, 24:1233–1265. 21

Harchaoui, Z. and Cappé, O. (2007). Retrospective mutiple change-point estimation with kernels. In *2007 IEEE/SP 14th Workshop on Statistical Signal Processing*, pages 768–772. IEEE. 38

Harchaoui, Z., Moulines, E., and Bach, F. R. (2009). Kernel change-point analysis. In *Advances in neural information processing systems*, pages 609–616. 38

Haynes, K., Fearnhead, P., and Eckley, I. A. (2017). A computationally efficient nonparametric approach for changepoint detection. *Statistics and Computing*, 27:1293–1305. 38

Healy, J. D. (1987). A note on multivariate cusum procedures. *Technometrics*, 29:409–412. 45

Hocking, T. D., Rigaill, G., Fearnhead, P., and Bourque, G. (2017). A log-linear time algorithm for constrained changepoint detection. *arXiv preprint arXiv:1703.03352*. 31

Hocking, T. D., Rigaill, G., Fearnhead, P., and Bourque, G. (2018). Generalized functional pruning optimal partitioning (gfpop) for constrained changepoint detection in genomic data. *arXiv preprint arXiv:1810.00117*. 31

Horváth, L. and Hušková, M. (2012). Change-point detection in panel data. *Journal of Time Series Analysis*, 33:631–648. 50, 171

Horváth, L., Kokoszka, P., and Reeder, R. (2013). Estimation of the mean of functional time series and a two-sample problem. *Journal of the Royal Statistical Society Series B*, 75:103–122. 21

Horváth, L., Kokoszka, P., and Rice, G. (2014). Testing stationarity of functional time series. *Journal of Econometrics*, 179:66–82. 21

Horváth, L., Kokoszka, P., and Steinebach, J. (1999). Testing for changes in multivariate dependent observations with an application to temperature changes. *Journal of Multivariate Analysis*, 68:96–119. 48

Hudson, D. J. (1966). Fitting segmented curves whose join points have to be estimated. *Journal of the American Statistical Association*, 61:1097–1129. 42

Inclan, C. (1993). Detection of multiple changes of variance using posterior odds. *Journal of Business and Economic Statistics*, 11:289–300. 38

Jamali, S., Jönsson, P., Eklundh, L., Ardö, J., and Seaquist, J. (2015). Detecting changes in vegetation trends using time series segmentation. *Remote Sensing of Environment*, 156:182–195. 29

James, B., James, K. L., and Siegmund, D. (1992). Asymptotic approximations for likelihood ratio tests and confidence regions for a change-point in the mean of a multivariate normal distribution. *Statistica Sinica*, 2:69–90. 45

James, G. M., Wang, J., and Zhu, J. (2009). Functional linear regression that's interpretable. *The Annals of Statistics*, 37:2083–2108. 24, 74

James, N. A., Kejariwal, A., and Matteson, D. S. (2016). Leveraging cloud data to mitigate user experience from 'breaking bad'. In *Big Data (Big Data), 2016 IEEE International Conference on*, pages 3499–3508. IEEE. 28

Jeng, X. J., Cai, T. T., and Li, H. (2012). Simultaneous discovery of rare and common segment variants. *Biometrika*, 100:157–172. 29

Ji, H. and Müller, H.-G. (2017). Optimal designs for longitudinal and functional data. *Journal of the Royal Statistical Society Series B*, 79:859–876. 21

Jirak, M. (2015). Uniform change point tests in high dimension. *The Annals of Statistics*, 43:2451–2483. 51, 52, 232

Joseph, L. (1989). *The multi-path change-point*. PhD thesis, McGill University Libraries. 46

Joseph, L., Vandal, A. C., and Wolfson, D. B. (1996). Estimation in the multipath change point problem for correlated data. *Canadian Journal of Statistics*, 24:37–53. 46

Joseph, L. and Wolfson, D. B. (1992). Estimation in multi-path change-point problems. *Communications in Statistics-Theory and Methods*, 21:897–913. 46

Joseph, L. and Wolfson, D. B. (1993). Maximum likelihood estimation in the multi-path change-point problem. *Annals of the Institute of Statistical Mathematics*, 45:511–530. 46, 47

Joseph, L., Wolfson, D. B., Du Berger, R., and Lyle, R. M. (1997). Analysis of panel data with change-points. *Statistica Sinica*, 7:687–703. 46

Keogh, E., Chu, S., Hart, D., and Pazzani, M. (2004). Segmenting time series: A survey and novel approach. In *Data mining in time series databases*, pages 1–21. World Scientific. 41, 129

Killick, R., Fearnhead, P., and Eckley, I. A. (2012). Optimal detection of changepoints with a linear computational cost. *Journal of the American Statistical Association*, 107:1590–1598. 31, 38

Kim, H.-J., Fay, M. P., Feuer, E. J., and Midthune, D. N. (2000). Permutation tests for joinpoint regression with applications to cancer rates. *Statistics in Medicine*, 19:335–351. 42

Kim, S.-J., Koh, K., Boyd, S., and Gorinevsky, D. (2009). ℓ_1 trend filtering. *SIAM review*, 51:339–360. 40, 128

Kirch, C., Muhsal, B., and Ombao, H. (2015). Detection of changes in multivariate time series with application to eeg data. *Journal of the American Statistical Association*, 110:1197–1216. 49

Kneip, A., Poß, D., and Sarda, P. (2016). Functional linear regression with points of impact. *The Annals of Statistics*, 44:1–30. 25, 59, 86

Kong, D., Xue, K., Yao, F., and Zhang, H. H. (2016). Partially functional linear regression in high dimensions. *Biometrika*, 103:147–159. 21, 28

Korkas, K. and Fryzlewicz, P. (2017). Multiple change-point detection for non-stationary time series using wild binary segmentation. *Statistica Sinica*, 27:287–311. 33

Krishnaiah, P., Miao, B., and Zhao, L. (1987). Local likelihood method in the problems related to change points. Technical report, Pittsburgh University. 45

Kristensen, D. (2010). Nonparametric filtering of the realized spot volatility: A kernel-based approach. *Econometric Theory*, 26:60–93. 85

Lavielle, M. and Moulines, E. (2000). Least-squares estimation of an unknown number of shifts in a time series. *Journal of Time Series Analysis*, 21:33–59. 30

Lavielle, M. and Teyssiere, G. (2006). Detection of multiple change-points in multivariate time series. *Lithuanian Mathematical Journal*, 46:287–306. 48

Lee, C.-B. (1995). Estimating the number of change points in a sequence of independent normal random variables. *Statistics and Probability Letters*, 25:241–248. 30

Lee, C.-B. (1997). Estimating the number of change points in exponential families distributions. *Scandinavian Journal of Statistics*, 24:201–210. 30

Li, H., Munk, A., and Sieling, H. (2016). Fdr-control in multiscale change-point segmentation. *Electronic Journal of Statistics*, 10:918–959. 31

Li, J., Xu, M., Zhong, P.-S., and Li, L. (2019). Change point detection in the mean of high-dimensional time series data under dependence. *arXiv preprint arXiv:1903.07006*. 43, 53

Lin, K., Sharpnack, J., Rinaldo, A., and Tibshirani, R. J. (2016). Approximate recovery in changepoint problems, from ℓ_2 estimation error rates. *arXiv preprint arXiv:1606.06746*. 152, 153, 226, 227, 228

Lin, K., Sharpnack, J. L., Rinaldo, A., and Tibshirani, R. J. (2017). A sharp error analysis for the fused lasso, with application to approximate changepoint screening. In *Advances in Neural Information Processing Systems*, pages 6884–6893. 41, 114

Lin, Z., Cao, J., Wang, L., and Wang, H. (2015). A smooth and locally sparse estimator for functional linear regression via functional scad penalty. *arXiv preprint arXiv:1510.08547*. 25

Lowry, C. A. and Montgomery, D. C. (1995). A review of multivariate control charts. *IIE transactions*, 27:800–810. 45

Lung-Yut-Fong, A., Lévy-Leduc, C., and Cappé, O. (2011a). Robust changepoint detection based on multivariate rank statistics. In *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3608–3611. IEEE. 48

Lung-Yut-Fong, A., Lévy-Leduc, C., and Cappé, O. (2011b). Robust retrospective multiple change-point estimation for multivariate data. In *2011 IEEE Statistical Signal Processing Workshop (SSP)*, pages 405–408. IEEE. 48

Ma, T. F. and Yau, C. Y. (2016). A pairwise likelihood-based approach for changepoint detection in multivariate time series models. *Biometrika*, 103:409–421. 48

Maboudou-Tchao, E. M. and Hawkins, D. M. (2013). Detection of multiple change-points in multivariate data. *Journal of Applied Statistics*, 40:1979–1995. 48

Maeng, H. (2019a). Detecting linear trend changes and point anomalies in data sequences: Simulation code. *URL https://github.com/hmaeng/trendsegment*. 128

Maeng, H. (2019b). Regularised forecasting via smooth-rough partitioning of the regression coefficients: Simulation code. *URL https://github.com/hmaeng/srp*. 74

Maeng, H. (2019c). Trend segmentation for high-dimensional time series: Simulation code. *URL https://github.com/hmaeng/HiTS*. 19, 192

Maeng, H. and Fryzlewicz, P. (2019). Detecting linear trend changes and point anomalies in data sequences. *arXiv preprint arXiv:1906.01939*. 157, 158

Maidstone, R., Fearnhead, P., and Letchford, A. (2017a). Detecting changes in slope with an l_0 penalty. *arXiv preprint arXiv:1701.01672*. 40, 129

Maidstone, R., Hocking, T., Rigaill, G., and Fearnhead, P. (2017b). On optimal multiple changepoint algorithms for large data. *Statistics and Computing*, 27:519–533. 31

Mason, R. L., Champ, C. W., Tracy, N. D., Wierda, S. J., and Young, J. C. (1997). Assessment of multivariate process control techniques. *Journal of Quality Technology*, 29:140–143. 45

Matteson, D. S. and James, N. A. (2014). A nonparametric approach for multiple change point analysis of multivariate data. *Journal of the American Statistical Association*, 109:334–345. 34, 48

Matteson, D. S., James, N. A., Nicholson, W. B., and Segalini, L. C. (2013). Locally stationary vector processes and adaptive multivariate modeling. In *Acoustics, Speech*

and Signal Processing (ICASSP), 2013 IEEE International Conference on, pages 8722–8726. IEEE. 29

McDonald, J. A. and Owen, A. B. (1986). Smoothing with split linear fits. *Technometrics*, 28:195–208. 42

McKeague, I. W. and Sen, B. (2010). Fractals with point impact in functional linear regression. *The Annals of Statistics*, 38:2559–2586. 25

McZgee, V. E. and Carleton, W. T. (1970). Piecewise regression. *Journal of the American Statistical Association*, 65:1109–1124. 42

Messer, M., Kirchner, M., Schiemann, J., Roeper, J., Neininger, R., and Schneider, G. (2014). A multiple filter test for the detection of rate changes in renewal processes with varying variance. *The Annals of Applied Statistics*, 8:2027–2067. 34

Moore, G. and Babij, M. (2017). Iceland's great frost winter of 1917/1918 and its representation in reanalyses of the twentieth century. *Quarterly Journal of the Royal Meteorological Society*, 143:508–520. 97

Müller, H.-G., Sen, R., and Stadtmüller, U. (2011). Functional data analysis for volatility. *Journal of Econometrics*, 165:233–245. 85, 86

Olshen, A. B., Venkatraman, E., Lucito, R., and Wigler, M. (2004). Circular binary segmentation for the analysis of array-based dna copy number data. *Biostatistics*, 5:557–572. 16, 29, 32

Ombao, H., Raz, J., Von Sachs, R., and Guo, W. (2002). The slex model of a non-stationary random process. *Annals of the Institute of Statistical Mathematics*, 54:171–200. 48

Ombao, H., Von Sachs, R., and Guo, W. (2005). Slex analysis of multivariate nonstationary time series. *Journal of the American Statistical Association*, 100:519–531. 48

Pan, J. and Chen, J. (2006). Application of modified information criterion to multiple change point problems. *Journal of Multivariate Analysis*, 97:2221–2241. 30

Perreault, L., Parent, E., Bernier, J., Bobee, B., and Slivitzky, M. (2000). Retrospective multivariate bayesian change-point analysis: A simultaneous single change in the mean of several hydrological sequences. *Stochastic Environmental Research and Risk Assessment*, 14:243–261. 45

Pignatiello Jr, J. J. and Runger, G. C. (1990). Comparisons of multivariate cusum charts. *Journal of Quality Technology*, 22:173–186. 45

Preuss, P., Puchstein, R., and Dette, H. (2015). Detection of multiple structural breaks in multivariate time series. *Journal of the American Statistical Association*, 110:654–668. 49

Pumo, B. (1998). Prediction of continuous time processes by c [0, 1]-valued autoregressive process. *Statistical Inference for Stochastic Processes*, 1:297–309. 70

Quandt, R. E. (1958). The estimation of the parameters of a linear regression system obeying two separate regimes. *Journal of the American Statistical Association*, 53:873–880. 39

Ramsay, J. O. and Silverman, B. W. (2005). *Functional Data Analysis*. New York: Springer-Verlag. 21, 22

Reiss, P. T., Goldsmith, J., Shang, H. L., and Ogden, R. T. (2017). Methods for scalar-on-function regression. *International Statistical Review*, 85:228–249. 22

Reno, R. (2008). Nonparametric estimation of the diffusion coefficient of stochastic volatility models. *Econometric Theory*, 24:1174–1206. 85

Rigaill, G. (2015). A pruned dynamic programming algorithm to recover the best segmentations with 1 to kmax change-points. *Journal de la Société Française de Statistique*, 156:180–205. 31

Rintoul, S., Chown, S., DeConto, R., England, M., Fricker, H., Masson-Delmotte, V., Naish, T., Siegert, M., and Xavier, J. (2018). Choosing the future of antarctica. *Nature*, 558:233–241. 142

Robbins, M. W., Lund, R. B., Gallagher, C. M., and Lu, Q. (2011). Changepoints in the north atlantic tropical cyclone record. *Journal of the American Statistical Association*, 106:89–99. 16, 28

Robinson, L. F., Wager, T. D., and Lindquist, M. A. (2010). Change point estimation in multi-subject fmri studies. *Neuroimage*, 49:1581–1592. 29, 47

Ruppert, D. (2002). Selecting the number of knots for penalized splines. *Journal of Computational and Graphical Statistics*, 11:735–757. 68

Safikhani, A. and Shojaie, A. (2017). Joint structural break detection and parameter estimation in high-dimensional non-stationary var models. *arXiv preprint arXiv:1711.07357*. 52

Schröder, A. L. and Ombao, H. (2019). Fresped: Frequency-specific change-point detection in epileptic seizure multi-channel eeg data. *Journal of the American Statistical Association*, 114:115–128. 49

Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics*, 6:461–464. 25, 30, 63, 140

Sen, A. K. and Srivastava, M. S. (1973). On multivariate tests for detecting change in mean. *Sankhyā: The Indian Journal of Statistics Series A*, 35:173–186. 44

Serreze, M. C. and Meier, W. N. (2018). The arctic's sea ice cover: trends, variability, predictability, and comparisons to the antarctic. *Annals of the New York Academy of Sciences*. 141, 142

Shin, H. (2009). Partial functional linear regression. *Journal of Statistical Planning and Inference*, 139:3405–3418. 28

Shin, H. and Lee, M. H. (2012). On prediction rate in partial functional linear regression. *Journal of Multivariate Analysis*, 103:93–106. 28

Siegmund, D., Yakir, B., and Zhang, N. R. (2011). Detecting simultaneous variant intervals in aligned sequences. *The Annals of Applied Statistics*, 5:645–668. 48

Slepian, D. (1962). The one-sided barrier problem for gaussian noise. *Bell System Technical Journal*, 41:463–501. 221, 222

Smith, A. and Cook, D. (1980). Straight lines with a change-point: a bayesian analysis of some renal transplant data. *Applied Statistics*, 29:180–189. 39

Soh, Y. S. and Chandrasekaran, V. (2017). High-dimensional change-point estimation: Combining filtering with convex optimization. *Applied and Computational Harmonic Analysis*, 43:122–147. 53

Son, Y. S. and Kim, S. W. (2005). Bayesian single change point detection in a sequence of multivariate normal observations. *Statistics*, 39:373–387. 45

Spiriti, S., Eubank, R., Smith, P. W., and Young, D. (2013). Knot selection for least-squares and penalized splines. *Journal of Statistical Computation and Simulation*, 83:1020–1036. 42, 129

Srivastava, M. and Worsley, K. J. (1986). Likelihood ratio tests for a change in the multivariate normal mean. *Journal of the American Statistical Association*, 81:199–204. 45

Stephens, D. (1994). Bayesian retrospective multiple-changepoint identification. *Journal of the Royal Statistical Society Series C*, 43:159–178. 38

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society Series B*, 58:267–288. 24

Tibshirani, R. J. (2014). Adaptive piecewise polynomial estimation via trend filtering. *The Annals of Statistics*, 42:285–323. 40

Tickle, S., Eckley, I., Fearnhead, P., and Haynes, K. (2018). Parallelisation of a common changepoint detection method. *arXiv preprint arXiv:1810.03591*. 31

Venkatraman, E. and Olshen, A. B. (2007). A faster circular binary segmentation algorithm for the analysis of array cgh data. *Bioinformatics*, 23:657–663. 32

Vert, J.-P. and Bleakley, K. (2010). Fast detection of multiple change-points shared by many signals using group lars. In *Advances in neural information processing systems*, pages 2343–2351. 48

Vostrikova, L. (1981). Detection of the disorder in multidimensional random-processes. *Doklady Akademii Nauk SSSR*, 259:270–274. 31, 32, 157

Wang, D., Yu, Y., and Rinaldo, A. (2017). Optimal covariance change point localization in high dimension. *arXiv preprint arXiv:1712.09912*. 53

Wang, T. and Samworth, R. J. (2018). High dimensional change point estimation via sparse projection. *Journal of the Royal Statistical Society Series B*, 80:57–83. 51, 52, 189, 192

Wang, Y. (1995). Jump and sharp cusp detection by wavelets. *Biometrika*, 82:385–397. 42

Wierda, S. J. (1994). Multivariate statistical process control—recent results and directions for future research. *Statistica Neerlandica*, 48:147–168. 45

Wilson, R. C., Nassar, M. R., and Gold, J. I. (2010). Bayesian online learning of the hazard rate in change-point problems. *Neural computation*, 22:2452–2476. 38

Wood, S. N. (2006). *Generalized additive models: an introduction with R*. CRC press. 67

Woodall, W. H. and Ncube, M. M. (1985). Multivariate cusum quality-control procedures. *Technometrics*, 27:285–292. 45

Worsley, K. (1983). Testing for a two-phase multiple regression. *Technometrics*, 25:35–42. 39

Worsley, K. (1986). Confidence regions and tests for a change-point in a sequence of exponential family random variables. *Biometrika*, 73:91–104. 45

Xia, Z. and Qiu, P. (2015). Jump information criterion for statistical inference in estimating discontinuous curves. *Biometrika*, 102:397–408. 42

Xie, Y. and Siegmund, D. (2013). Sequential multi-sensor change-point detection. In *2013 Information Theory and Applications Workshop (ITA)*, pages 1–20. IEEE. 52

Yao, Y.-C. (1988). Estimating the number of change-points via schwarz'criterion. *Statistics and Probability Letters*, 6:181–189. 30

Yao, Y.-C. and Au, S.-T. (1989). Least-squares estimation of a step function. *Sankhyā: The Indian Journal of Statistics Series A*, 51:370–381. 30

Yu, B., Barrett, M. J., Kim, H.-J., and Feuer, E. J. (2007). Estimating joinpoints in continuous time scale for multiple change-point models. *Computational Statistics and Data Analysis*, 51:2420–2427. 42

Yuan, M. and Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society Series B*, 68:49–67. 48

Zhang, N. R. and Siegmund, D. O. (2007). A modified bayes information criterion with applications to the analysis of comparative genomic hybridization data. *Biometrics*, 63:22–32. 30

Zhang, N. R., Siegmund, D. O., Ji, H., and Li, J. Z. (2010). Detecting simultaneous changepoints in multiple sequences. *Biometrika*, 97:631–645. 43, 50

Zhou, J. and Chen, M. (2012). Spline estimators for semi-functional linear model. *Statistics and Probability Letters*, 82:505–513. 28

Zhou, J., Chen, Z., and Peng, Q. (2016). Polynomial spline estimation for partial functional linear regression models. *Computational Statistics*, 31:1107–1129. 28

Zhou, J., Wang, N.-Y., and Wang, N. (2013). Functional linear model with zero-value coefficient function at sub-regions. *Statistica Sinica*, 23:25–50. 25

Zhu, H., Yao, F., and Zhang, H. H. (2014). Structured functional additive regression in reproducing kernel hilbert spaces. *Journal of the Royal Statistical Society Series B*, 76:581–603. 21